ECMA

EUROPEAN COMPUTER MANUFACTURERS ASSOCIATION

STANDARD ECMA-6

for a

7 BIT INPUT/OUPUT CODED CHARACTER SET

AMAH 311/3

3rd Edition — July 1970

BRIEF HISTORY

The Technical Committee TC1 of ECMA met for the first time in December 1960 to prepare standard codes for Input/Output purposes. On April 30, 1965, Standard ECMA-6 was adopted by the General Assembly of ECMA.

Subsequent international activities necessitated the revision of ECMA-6, which was issued in June 1967. The present 3rd Edition does not differ essentially from the 2nd Edition. It has been improved particularly with regard to the definition of the characters of the Code Table. Furthermore, another table has been added, showing the International Reference Version of the Code Table.

THIS 3RD EDITION SUPERSEDES THE VERSION DATED JUNE 1967.

INTRODUCTION

1. In the preparation of the 7-bit coded character set particular importance has been attached to the integration of computers and communication systems. The aim has been to establish compatibility of communication oriented systems and of computer oriented systems. For this there is a need for one code which is satisfactory for such combined use, without necessarily being optimized for either one to the detriment of the other.

2. This Standard essentially consists of a table defining the character set and its coded representation, complemented by notes and legend. Furthermore, a second table is included showing the international version derived from the more general specification. The coded character set may be regarded as an alphabet in the abstract sense.

3. This basic alphabet may be implemented in various media with logically related codes as appropriate. For example this could include punched tape, punched cards, magnetic tape and transmission channels. This permits interchange of data to take place either indirectly by means of an intermediate recording in a physical medium, or by local electrical connection of various units (such as input and output devices and computers) or by means of data transmission equipment.

4. The implementation of this coded character set in physical media and for transmission, taking into account the need for error checking, is the subject of other standards ¤.

¤ NOTE: See:

ECMA-5 :	Data Interchange on 7 Track Magnetic Tape
ECMA-10 :	Data Interchange on Punched Tape
ECMA-12 :	Data Interchange on 9 Track Magnetic Tape at 31,5 bit per mm
ECMA-16 :	Basic Mode Control Procedures for Data Communication Systems using the ECMA 7 Bit Code
ECMA-20 :	Implementation of the ECMA 7 Bit Coded Character Set on Punched Cards
ECMA-23 :	Keyboards generating the Code Combinations of the Characters of the ECMA 7 Bit Coded Character Set
ECMA-24 :	Code Independent Information Transfer

ECMA 7 BIT CODED CHARACTER SET
FOR DATA INTERCHANGE

b_7					0	0	0	0	1	1	1	1
b_6					0	0	1	1	0	0	1	1
b_5					0	1	0	1	0	1	0	1
Bits b_4	b_3	b_2	b_1	Row \ Column	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	TC$_7$(DLE)	SP	0	③ (@)	P	` ④	p
0	0	0	1	1	TC$_1$ (SOH)	DC$_1$	!	1	A	Q	a	q
0	0	1	0	2	TC$_2$ (STX)	DC$_2$	" ⑥	2	B	R	b	r
0	0	1	1	3	TC$_3$ (ETX)	DC$_3$	② £ (#)	3	C	S	c	s
0	1	0	0	4	TC$_4$ (EOT)	DC$_4$	② \$ (¤)	4	D	T	d	t
0	1	0	1	5	TC$_5$ (ENQ)	TC$_8$(NAK)	%	5	E	U	e	u
0	1	1	0	6	TC$_6$ (ACK)	TC$_9$(SYN)	&	6	F	V	f	v
0	1	1	1	7	BEL	TC$_{10}$(ETB)	' ⑥	7	G	W	g	w
1	0	0	0	8	FE$_0$ (BS)	CAN	(	8	H	X	h	x
1	0	0	1	9	FE$_1$ (HT)	EM	)	9	I	Y	i	y
1	0	1	0	10	FE$_2$ (LF)	SUB	*	: ⑦	J	Z	j	z
1	0	1	1	11	FE$_3$ (VT)	ESC	+	; ⑦	K	③ ([)	k	③ ({)
1	1	0	0	12	FE$_4$ (FF)	IS$_4$ (FS)	, ⑥	<	L	③ (\)	l	③ (|)
1	1	0	1	13	FE$_5$ (CR) ①	IS$_3$ (GS)	-	=	M	③ (])	m	③ (})
1	1	1	0	14	SO	IS$_2$ (RS)	.	>	N	^ ④ ⑥	n	‾ ④ ⑤
1	1	1	1	15	SI	IS$_1$ (US)	/	?	O	_	o	DEL

TABLE 1

See Notes on page 4 and 5

INTERNATIONAL REFERENCE VERSION OF THE ECMA 7 BIT CODED CHARACTER SET

Bits												
				b7	0	0	0	0	1	1	1	1
				b6	0	0	1	1	0	0	1	1
				b5	0	1	0	1	0	1	0	1
b4	b3	b2	b1	Column / Row	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	TC$_7$(DLE)	SP	0	@	P	`	p
0	0	0	1	1	TC$_1$ (SOH)	DC$_1$	!	1	A	Q	a	q
0	0	1	0	2	TC$_2$ (STX)	DC$_2$	"	2	B	R	b	r
0	0	1	1	3	TC$_3$ (ETX)	DC$_3$	#	3	C	S	c	s
0	1	0	0	4	TC$_4$ (EOT)	DC$_4$	¤	4	D	T	d	t
0	1	0	1	5	TC$_5$ (ENQ)	TC$_8$(NAK)	%	5	E	U	e	u
0	1	1	0	6	TC$_6$ (ACK)	TC$_9$(SYN)	&	6	F	V	f	v
0	1	1	1	7	BEL	TC$_{10}$(ETB)	'	7	G	W	g	w
1	0	0	0	8	FE$_0$ (BS)	CAN	(	8	H	X	h	x
1	0	0	1	9	FE$_1$ (HT)	EM	)	9	I	Y	i	y
1	0	1	0	10	FE$_2$ (LF)	SUB	*	:	J	Z	j	z
1	0	1	1	11	FE$_3$ (VT)	ESC	+	;	K	[	k	{
1	1	0	0	12	FE$_4$ (FF)	IS$_4$ (FS)	,	<	L	\	l	|
1	1	0	1	13	FE$_5$ (CR)	IS$_3$ (GS)	–	=	M	]	m	}
1	1	1	0	14	SO	IS$_2$ (RS)	.	>	N	↑	n	~
1	1	1	1	15	SI	IS$_1$ (US)	/	?	O	_	o	DEL

TABLE 2

This version is based on Table 1 and is intended for general use in all cases when there is no requirement to use a national version or any other "application oriented" character set.

NOTES ABOUT THE TABLES

① The format Effectors are intended for equipment in which horizontal and vertical movements are effected separately. If equipment requires the action of Carriage Return to be combined with a vertical movement, the Format Effector for that vertical movement may be used to effect the combined movements. For example, if New Line (CR + LF) is required, FE_2 will be used to represent it. This requires agreement between the sender and the recipient of the data.

The use of these combined functions is not permitted in public telecommunications facilities.

② The symbol £ is assigned to position 2/3 and the symbol $ is assigned to position 2/4. In a situation where there is no requirement for the symbol £ the symbol # (number symbol) may be used in position 2/3 and where there is no requirement for the symbol $ the symbol ¤ (sun symbol) may be used in position 2/4. It should be noted that, unless otherwise agreed between sender and recipient, the symbols £, $ or ¤ do not designate the currency of a specific country.

③ Reserved for National Use. These positions are primarily intended for alphabetic extension. If they are not required for that purpose or if they are to be used for international interchange, the symbols shown in parenthesis must be used.

④ Position 5/14, 6/0 and 7/14 provide for the diacritical signs "circumflex accent", "grave accent" and "overline". However, these positions may be used for other graphical characters when it is necessary to have 8, 9 or 10 positions for national use.

⑤ Position 7/14 is used for the graphical symbol ‾ (overline), the graphical representation of which may vary according to national use to represent ˜ (tilde) or another diacritical sign provided that there is no risk of confusion with another graphical symbol included in the table.

⑥ The graphics in positions 2/2, 2/7, 2/12, 5/14 have respectively the significance of "quotation mark", "apostrophe", "comma" and "upward arrow"; however, these characters take on the significance of "diaresis", "acute accent", "cedilla" and "circumflex accent" diacritical signs when they are preceded by "Backspace". (see Explanatory Note 2, page 8)

⑦ If 10 and 11 as single characters are needed (for example, for Sterling currency subdivision), they should take the place of "colon" (:) and "semi-colon"(;) respectively. These substitutions require agreement between the sender and the recipient of the data. 10 and 11 as single characters are not permitted in public telecommunications facilities.

LEGEND

1. CONTROL CHARACTERS

Abbreviation	Note	Meaning	Position in the Code Table
ACK		Acknowledge	0/6
BEL		Bell	0/7
BS		Backspace	0/8
CAN		Cancel	1/8
CR	1	Carriage Return	0/13
DC		Device Control	-
DEL		Delete	7/15
DLE		Data Link Escape	1/0
EM		End of Medium	1/9
ENQ		Enquiry	0/5
EOT		End of Transmission	0/4
ESC		Escape	1/11
ETB		End of Transmission Block	1/7
ETX		End of Text	0/4
FE		Format Effector	-
FF		Form Feed	0/12
FS		File Separator	1/12
GS		Group Separator	1/13
HT		Horizontal Tabulation	0/9
IS		Information Separator	-
LF		Line Feed	0/10
NAK		Negative Acknowledge	1/5
NUL		Null	0/0
RS		Record Separator	1/14
SI		Shift-In	0/15
SO		Shift-Out	0/14
SOH		Start of Heading	0/1
SP		Space	2/0
SUB		Substitute Character	1/10
STX		Start of Text	0/2
SYN		Synchronous Idle	1/6
TC		Transmission Control	-
US		Unit Separator	1/15
VT		Vertical Tabulation	0/11

See note on page 4.

2. GRAPHIC CHARACTERS

Graphic	Note	Name	Position in the Code Table
(space)		A normally non-printing graphic character	2/0
!		Exclamation mark	2/1
"	6	Quotation mark, Diaresis	2/2
£	2	Pound symbol	2/3
#	2	Number symbol	2/3
$	2	Dollar symbol	2/4
¤	2	Sun symbol	2/4
%		Percent	2/5
&		Ampersand	2/6
'	6	Apostrophe, Closing single quotation mark, Acute Accent	2/7
(		Opening Parenthesis	2/8
)		Closing Parenthesis	2/9
*		Asterisk	2/10
+		Plus sign	2/11
,	6	Comma, Cedilla	2/12
-		Hyphen, Minus sign	2/13
.		Full stop (period)	2/14
/		Solidus	2/15
:	7	Colon	3/10
;	7	Semi-colon	3/11
<		Less than	3/12
=		Equals	3/13
>		Greater than	3/14
?		Question mark	3/15
@	3	Commercial at	4/0
A			4/1
⋮		} Upper Case alphabet	⋮
Z			5/10
[	3	Opening square bracket	5/11
\	3	Reverse solidus	5/12
]	3	Closing square bracket	5/13
^	4 6	Upward arrow, Circumflex accent	5/14
_		Underline	5/15
`	4	Grave accent, Opening single quotation mark	6/0
a			6/1
⋮		} Lower Case alphabet	⋮
z			7/10
{	3	Opening brace	7/11
|	3	Vertical line	7/12
}	3	Closing brace	7/13
‾	4 5	Overline, Tilde	7/14

See notes on page 4 and 5.

EXPLANATORY NOTES

1. Numbering of the position in the Code Table

Within any one character the bits are identified by b_7, b_6, ... b_1, where b_7 is the highest order, or most significant bit, and b_1 is the lowest order, or least significant bit.

If desired these may be given a numerical significance in the binary system, thus:

Bit identification	b_7	b_6	b_5	b_4	b_3	b_2	b_1
Significance	64	32	16	8	4	2	1

In the code table the columns and rows are identified by numbers written in binary and decimal notations.

Any one position in the code table may be identified either by its bit pattern, or by its column and row numbers. For instance, the position containing the figure 1 in the table may be identified:

- by its bit pattern, e.g. 011 0001
- by its column and row numbers, e.g. 3/1

2. Diacritical signs (positions: 2/2, 2/7, 2/12, 5/14, 6/0, 7/14)

In the character set, some printing symbols may be designed to permit their use for the composition of accented letters or composite characters when necessary for general interchange of information. A sequence of three characters, comprising a letter, "Backspace" and one of these symbols, is needed for this composition, and the symbol is then regarded as a diacritical sign. It should be noted that these symbols take on their diacritical significance only when they follow the "Backspace" character; for example, the symbol corresponding to the code combination 2/7 (') normally has the significance of "apostrophe", but becomes the diacritical sign "acute accent" when preceded by the "Backspace" character.

In order to increase efficiency, it is possible to introduce accented letters (as single character) in the positions marked by Notes 3 and 4 in the code table. According to national requirements, these positions may alternatively contain special diacritical signs, such as Haček (ˇ), ring (°), etc.

3. Meaning of the Characters

The meaning of the graphics is not defined by this Standard. It will be necessary to reach agreement on the meaning and this will depend upon the particular application except in cases where other standards exist. However, no interpretation may be chosen which is contradictory to the customary meaning. A graphical symbol can have more than one meaning, e.g. the graphical symbol - (minus) also can have the meanings of hyphen or separation mark. The font design of the characters is not part of this Standard.

4. Uniqueness of Character Allocation (Table 1)

A character allocated to a position in the code table may not be placed elsewhere in the table. For example, in the case of position 2/3 the character not used cannot be placed elsewhere. This applies also to positions showing a graphic in parenthesis. If such a position is filled with a national character, the graphic in parenthesis cannot be reallocated to another code position.

FUNCTIONAL CHARACTERISTICS RELATED TO CONTROL CHARACTERS AND OTHER NON GRAPHIC CHARACTERS.

The definitions in this section are normally stated in general terms and more explicit definitions may be needed for specific implementations of the code table.

Further studies of the various fields of application will lead to agreements on more precise specifications.

1. Classes of Controls

Some controls fall naturally into classes and these have been given class names. The names and description of the classes are as follows:

TC Transmission Control

A functional character intended to control or facilitate transmission of information over telecommunication networks.

FE Format Effector

A functional character mainly intended for the control of the layout and positioning of information on printing and/or display devices. In the definitions of specific Format Effectors in Section 2 any reference to printing devices should be interpreted as including display devices.

The definitions of Format Effectors use the following concept:

i) The printed page is composed of a number of lines of characters.

ii) The characters forming a line occupy a number of positions called Character Positions.

iii) The Active Position is that character position in which the character about to be processed would appear, if it were to be printed. The Active Position normally advances one character position at a time.

DC Device Control

A functional character for the control of a local or remote ancillary device (or devices) connected to a data processing and/or telecommunication system. This control is not intended to control the telecommunications system; this control should be achieved by the use of TCs.
Certain preferred uses of the individual DCs are given in Section 2.

IS Information Separator

A character which is used to separate and qualify data logically. There are four such characters. They may be used either in hierarchical order or non-hierarchically; in the latter case their specific meaning depends on their applications.
If they are used hierarchically the ascending order is:

US, RS, GS, FS.

In this case data normally delimited by a particular separator cannot be split by a higher order separator but will be considered as delimited by any higher order separator.

2. Specific Controls and other non graphic characters

Individual members of the classes of controls are sometimes referred to by abbreviated class name and a subscript number (e.g., TC_5) and sometimes by a specific name indicative of their use (e.g., ENQ). These controls, those which do not fall into classes and other non graphic characters are defined below.

Different but related meanings may be associated with some of the controls but this would normally require agreement between sender and recipient of the data.

ACK Acknowledge

A Transmission Control character transmitted by a receiver as an affirmative response to the sender.

BEL Bell

A character for use when there is a need to call for attention; it may control alarm or attention devices.

BS Backspace

A format Effector which moves the active position one character position backwards on the same line.

CAN Cancel

The first character of a two character sequence indicating that the data associated with it is in error or is to be ignored.
This data includes the CAN sequence itself and all data previous to it up to, but excluding, the nearest previous occurence of the character following CAN. A CAN sequence is always to be interpreted as if any previously occuring CAN sequence had already been completely processed.

In certain circumstances some control character may not be permitted to follow CAN or there may be a limit to the amount or type of data able to be cancelled.

Examples:

1) A B X C D E X G $\underset{N}{\overset{C}{A}}$ E F G yields A B X C D E F G

2) A B X C D E X B $\underset{N}{\overset{C}{A}}$ E $\underset{N}{\overset{C}{A}}$ B C D yields A B C D

3) A B C $\underset{N}{\overset{C}{A}}$ C D E yields A B C D E

CR Carriage Return

A Format Effector which moves the active position to the first character position on the same line.

DC1 A Device Control character which is primarily intended for turning on or starting an ancillary device.
If it is not required for this purpose, it may be used to restore a device to the basic mode of operation (see also DC2 and DC3), or for any other device control function not provided by other DCs.

DC2 A Device Control character which is primarily intended for turning on or starting an ancillary device.
If it is not required for this purpose, it may be used to set a device to a special mode of operation (in which case DC1 is used to restore the device to the basic mode), or for any other device control function not provided by other DCs.

DC3 A Device Control character which is primarily intended for turning off or stopping an ancillary device. This function can be a secondary level stop e.g. wait, pause, stand-by or halt (in which case DC1 is used to restore normal operation). If it is not required for this purpose, it may be used for any other device control function not provided by other DCs.

DC4 A Device Control character which is primarily intended for turning off, stopping or interrupting an ancillary device.
If it is not required for this purpose, it may be used for any other device control function not provided by other DCs.

Examples of use of the Device Controls:

1) One switching		
	on - DC2	off - DC4
2) Two independent switchings		
First one	on - DC2	off - DC4
Second one	on - DC1	off - DC3
3) Two dependent switchings		
general	on - DC2	off - DC4
particular	on - DC1	off - DC3
4) Input and output switching		
Output	on - DC2	off - DC4
Input	on - DC1	off - DC3

DEL Delete

A character used primarily to erase or obliterate erroneous or unwanted character in punched tape. DEL characters may also serve to accomplish media-fill or time-fill. They may be inserted into or removed from a stream of data without affecting the information content of that stream but then the addition or removal of these characters may affect the information layout and/or the control of equipment.

DLE Data Link Escape

A Transmission Control character which will change the meaning of a limited number of contiguously following characters. It is used exclusively to provide supplementary data transmission control functions. Only graphics and transmission control characters can be used in DLE sequences.

EM End of Medium

A control character which may be used to identify the physical end of a medium, or the end of the used, or wanted portion of information recorded on a medium. The position of this character does not necessarily correspond to the physical end of the medium.

ENQ Enquiry

A Transmission Control character used as a request for a response from a remote station - the response may include station identification and/or station status. When a "Who are you" function is required on the general switched network, the first use of ENQ after the connection is established shall have the meaning "Who are you" (station identification). Subsequent use of ENQ may, or may not, include the function "Who are you", as determined by agreement.

EOT End of Transmission

A Transmission Control character used to indicate the conclusion of the transmission of one or more texts.

ESC Escape

A functional character which may be used to extend the standard character set of the code. It is a warning or non locking shift character which changes the meaning of a limited number of contiguously following code combinations. The use of such a sequence of characters is specified in Standard ECMA- ¤.

Escape sequences are used primarily to obtain additional control functions which may provide amongst other things graphic sets outside the standard set. Such control functions must not be used as additional Transmission Controls.

CAN, DEL, SI, SO, SUB and the ten Transmission Controls must not be used in defining Escape sequences.

ETB End of Transmission Block

A Transmission Control character used to indicate the end of a transmission block of data where data is divided into such blocks for transmission purposes.

ETX End of Text

A Transmission Control character which terminates a text.

FF Form Feed

A Format Effector which advances the active position to the same character position on a predetermined line of the next form or page.

HT Horizontal Tabulation

A Format Effector which advances the active position to the next predetermined character position on the same line,

¤ (Presently Draft Standard TC1/70/11)

IS_1 (US): A character used to separate and qualify data logically; its specific meaning has to be defined for each application. If this character is used in hierarchical order as specified in the general definition of IS, it delimits a data item called a UNIT.

IS_2 (RS): A character used to separate and qualify data logically; its specific meaning has to be defined for each application. If this character is used in hierarchical order as specified in the general definition of IS, it delimits a data item called a RECORD.

IS_3 (GS): A character used to separate and qualify data logically; its specific meaning has to be defined for each application. If this character is used in hierarchical order as specified in the general definition of IS, it delimits a data item called a GROUP.

IS_4 (FS): A character used to separate and qualify data logically; its specific meaning has to be defined for each application. If this character is used in hierarchical order as specified in the general definition of IS, it delimits a data item called a FILE.

LF Line Feed

A Format Effector which advances the active position to the same character position on the next line.

NAK Negative Acknowledge

A Transmission Control character transmitted by a receiver as a negative response to the sender.

NUL Null

A character used to accomplish media-fill or time-fill. NUL characters may be inserted into or removed from a stream of data without affecting the information content of that stream but then the addition or removal of these characters may affect the information layout and/or the control of equipment.

SI Shift-In ¤

The shift-in character means that the code combinations which follow shall be interpreted according to the standard code table.

SO Shift-Out ¤

The shift-out character means that the code combinations which follow shall be interpreted as outside the standard code table until a shift-in character is reached. However, all the control characters (columns 0 and 1) and Delete will retain their standard interpretation.

¤ NOTE: The effect of this character when using code extension techniques is described in Standard ECMA- * (presently Draft Standard TC1/70/11).

The shift-out character is limited to extension of the graphics.

SOH Start of Heading

A Transmission Control character used as the first character of a heading of an information message.

SP Space (see also chapter 2, Graphic Symbols, page 6)

A layout character which advances the active position one character position on the same line.

SUB Substitute Character

A character used to replace a character which has been found to be invalid or in error. SUB is introduced by the processing or transmission system.

STX Start of Text

A Transmission Control character which precedes a text and which is used to terminate a heading.

SYN Synchronous Idle

A Transmission Control character used by a synchronous transmission system in the absence of any other character (idle condition) to provide a signal from which synchronism may be achieved or retained between terminal equipments.

VT Vertical Tabulation

A Format Effector which advances the active position to the same character position on the next predetermined line.

IBM

A Programmer's Introduction to IBM System/360 Assembler Language

International Business Machines Corporation

Student Text

QA76
.8
.I12 I61
c.5
Sci

UNIVERSITY
OF CHICAGO
LIBRARY

res

Minor Revision (August 1970)

This publication is a minor revision of Form SC20-1646-5 incorporating corrections made on pages dated (8/70). The original publication is not obsoleted.

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality. Address comments concerning the contents of this publication to IBM Corporation, DPD Education Development - Publications Services, Education Center, South Road, Poughkeepsie, New York 12602.

© Copyright International Business Machines Corporation 1956, 1969

Preface

This student text is an introduction to System/360 assembler language coding. It provides many examples of short programs shown in assembled form. Some elementary programming techniques and the specific instructions illustrated in the programs are discussed in simple, relatively nontechnical terms. Much of the text is based on information in *IBM System/360 Principles of Operation* (GA22-6821). This includes a brief review of relevant System/360 concepts and descriptions of selected assembler language instructions for arithmetic, logical, and branching operations. Standard (fixed-point), decimal, and floating-point arithmetic are discussed. The book also includes an elementary introduction to assembler language and the assembler program, and chapters on base register addressing and on program linkages and relocation. The coding of many other common programming techniques, such as the use of branches, loops, and counters, is shown. The use of macro instructions is demonstrated, but not covered in detail. Program flowcharting and input/output operations are beyond the scope of the book.

The publication is a sampler rather than a comprehensive textbook. It is intended for supplementary reading for the student in a regular course of study on System/360 assembler language coding, and for the novice programmer. In general, the reader will find that the program examples are quite simple at the beginning of each chapter, or major subject division, and become progressively more complex. If the going seems difficult, it is suggested that he simply skip to the next subject and come back later.

The student should have access to two IBM System/360 System Reference Library (SRL) manuals for reference purposes: the *Principles of Operation* and the assembler specification manual for one of the System/360 operating systems. (All publications and their form numbers are listed at the end of the Preface.) He should also be familiar with fundamental concepts of data processing and the basic operating principles of System/360. Two IBM programmed instruction (P.I.) courses, or their equivalent, are prerequisite to a full understanding of this student text: *Computing System Fundamentals* and *Introduction to System/360*. The student who is not enrolled in a comprehensive programming course will find the P.I. book *Fundamentals of Programming* a valuable guide to problem analysis and program flowcharting.

The text and programs of this book have been revised throughout, mainly to reflect changes in programming conventions attributable to the development of System/360 operating systems. Chapter 1 is new, and several sections in other chapters have been entirely rewritten. The sample programs have been reassembled under the widely used Disk Operating System (DOS). As far as possible, usages limited to DOS have been avoided, and the programs and text in general are applicable to System/360 models 25, 30, 40, 50, 65, and 75, under any of the operating systems.

IBM publications that may be useful to the student are:

IBM System/360 Principles of Operation (SRL manual GA22-6821)

IBM System/360 Reference Data (card GX20-1703)

IBM System/360 System Summary (SRL manual GA22-6810)

Number Systems (Student Text GC20-1618)

Introduction to IBM System/360 Architecture (Student Text GC20-1667)

Introduction to System/360 (P.I. Course GR29-0256 through -0259)

Computing System Fundamentals (P.I. Course GR29-0280 through -0282)

Fundamentals of Programming P.I. Course SR29-0019)

System/360 Assembler Language Coding (P.I. Course SR29-0231 through -0235)

The form numbers of the assembler specification manuals for the various System/360 programming systems are:

Basic Programming Support (Tape System)–GC24-3335
Basic Operating System–GC24-3361
Tape Operating System }
Disk Operating System } GC24-3414
Operating System–GC28-6514

CONDITION CODE SETTINGS

Code State	0	1	2	3
Mask Bit Position	8	4	2	1
Fixed-Point Arithmetic				
Add H/F	zero	< zero	>zero	overflow
Add Logical	zero no carry	not zero no carry	zero carry	not zero carry
Compare H/F (A:B)	equal	A low	A high	--
Load and Test	zero	<zero	> zero	carry
Load Complement	zero	<zero	> zero	overflow
Load Negative	zero	<zero	--	--
Load Positive	zero	--	> zero	overflow
Shift Left Double	zero	<zero	> zero	overflow
Shift Left Single	zero	<zero	> zero	overflow
Shift Right Double	zero	<zero	> zero	--
Shift Right Single	zero	<zero	> zero	--
Subtract H/F	zero	<zero	> zero	overflow
Subtract Logical	--	not zero no carry	zero carry	not zero carry
Decimal Arithmetic				
Add Decimal	zero	< zero	> zero	overflow
Compare Decimal (A:B)	equal	A low	A high	--
Subtract Decimal	zero	< zero	> zero	overflow
Zero and Add	zero	<zero	> zero	overflow
Logical Operations				
And	zero	not zero	--	--
Compare Logical (A:B)	equal	A low	A high	--
Edit	zero	<zero	> zero	--
Edit and Mark	zero	<zero	> zero	--
Exclusive Or	zero	not zero	--	--
Or	zero	not zero	--	--
Test Under Mask	zero	mixed	--	one(s)
Translate and Test	zero	incomplete	complete	--

EXTENDED MNEMONIC CODES FOR THE BRANCH ON CONDITION INSTRUCTION

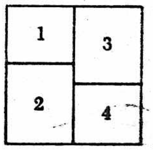
Assembler Code		Meaning	Machine Instruction Generated	
B	D2(X2,B2)	Branch Unconditional	BC	15,D2(X2,B2)
BR	R2	Branch Unconditional (RR format)	BCR	15,R2
NOP	D2(X2,B2)	No Operation	BC	0,D2(X2,B2)
NOPR	R2	No Operation (RR format)	BCR	0,R2
	Used after compare instructions (A:B)			
BH	D2(X2,B2)	Branch on High	BC	2,D2(X2,B2)
BL	D2(X2,B2)	Branch on Low	BC	4,D2(X2,B2)
BE	D2(X2,B2)	Branch on Equal	BC	8,D2(X2,B2)
BNH	D2(X2,B2)	Branch on Not High	BC	13,D2(X2,B2)
BNL	D2(X2,B2)	Branch on Not Low	BC	11,D2(X2,B2)
BNE	D2(X2,B2)	Branch on Not Equal	BC	7,D2(X2,B2)
	Used after arithmetic instructions			
BO	D2(X2,B2)	Branch on Overflow	BC	1,D2(X2,B2)
BP	D2(X2,B2)	Branch on Plus	BC	2,D2(X2,B2)
BM	D2(X2,B2)	Branch on Minus	BC	4,D2(X2,B2)
BZ	D2(X2,B2)	Branch on Zero	BC	8,D2(X2,B2)
BNP	D2(X2,B2)	Branch on Not Plus	BC	13,D2(X2,B2)
BNM	D2(X2,B2)	Branch on Not Minus	BC	11,D2(X2,B2)
BNZ	D2(X2,B2)	Branch on Not Zero	BC	7,D2(X2,B2)
	Used after Test under Mask instructions			
BO	D2(X2,B2)	Branch if Ones	BC	1,D2(X2,B2)
BM	D2(X2,B2)	Branch if Mixed	BC	4,D2(X2,B2)
BZ	D2(X2,B2)	Branch if Zeros	BC	8,D2(X2,B2)
BNO	D2(X2,B2)	Branch if Not Ones	BC	14,D2(X2,B2)

The University of Chicago Library

EBCDIC CHART

The 256-position chart at the right, outlined by the heavy black lines, shows the graphic characters and control character representations for the Extended Binary-Coded Decimal Interchange Code (EBCDIC). The bit-position numbers, bit patterns, hexadecimal representations and card hole patterns for these and other possible EBCDIC characters are also shown.

To find the card hole patterns for most characters, partition the chart into four blocks as follows:



- Block 1: Zone punches at top of table; digit punches at left
- Block 2: Zone punches at bottom of table; digit punches at left
- Block 3: Zone punches at top of table; digit punches at right
- Block 4: Zone punches at bottom of table; digit punches at right

Fifteen positions, indicated by circled numbers, are exceptions to the above arrangement. The card hole patterns for these positions are given below the chart.

Following are some examples of the use of the EBCDIC chart:

Character	Type	Bit Pattern	Hex	Hole Pattern	
				Zone Punches	Digit Punches
PF	Control Character	00 00 0100	04	12 - 9	- 4
%	Special Graphic	01 10 1100	6C	0	- 8 - 4
R	Upper Case	11 01 1001	D9	11	- 9
a	Lower Case	10 00 0001	81	12 - 0	- 1
	Control Character, function not yet assigned	00 11 0000	30	12 - 11 - 0 - 9	- 8 - 1

Bit Positions
01 23 4567

EBCDIC CHART

			00				01				10				11				Bit Positions 0,1
			00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11	Bit Positions 2,3
			0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	First Hexadecimal Digit
			12				12	12		12	12	12		12	12				Zone Punches
				11				11	11	11		11	11	11		11			
					0		0		0	0	0		0	0			0		
Bit Positions 4, 5, 6, 7	Second Hexadecimal Digit	Digit Punches	9	9	9	9	9	9	9	9									Digit Punches
0000	0	8-1	① NUL	② DLE	DS ③	④	SP ⑤	& ⑥	- ⑦	⑧					⑨	⑩	⑪	⑫ 0	8-1
0001	1	1	SOH	DC1	SOS				⑬ /		a	j			A	J	⑭	1	1
0010	2	2	STX	DC2	FS	SYN					b	k	s		B	K	S	2	2
0011	3	3	ETX	TM							c	l	t		C	L	T	3	3
0100	4	4	PF	RES	BYP	PN					d	m	u		D	M	U	4	4
0101	5	5	HT	NL	LF	RS					e	n	v		E	N	V	5	5
0110	6	6	LC	BS	ETB	UC					f	o	w		F	O	W	6	6
0111	7	7	DEL	IL	ESC	EOT					g	p	x		G	P	X	7	7
1000	8	8		CAN							h	q	y		H	Q	Y	8	8
1001	9	8-1		EM							i	r	z		I	R	Z	9	9
1010	A	8-2	SMM	CC	SM		¢	!	⑮	:									8-2
1011	B	8-3	VT	CU1	CU2	CU3	.	$	,	#									8-3
1100	C	8-4	FF	IFS		DC4	<	*	%	@									8-4
1101	D	8-5	CR	IGS	ENQ	NAK	(	)	_	'									8-5
1110	E	8-6	SO	IRS	ACK		+	;	>	=									8-6
1111	F	8-7	SI	IUS	BEL	SUB	|	¬	?	"									8-7
			12				12				12	12		12	12	12		12	Zone Punches
				11				11				11	11	11		11	11	11	
					0				0		0		0	0	0		0	0	
			9	9	9	9									9	9	9	9	

Card Hole Patterns (exceptions to punches shown in chart)

① 12-0-9-8-1
② 12-11-9-8-1
③ 11-0-9-8-1
④ 12-11-0-9-8-1
⑤ No Punches
⑥ 12
⑦ 11
⑧ 12-11-0
⑨ 12-0
⑩ 11-0
⑪ 0-8-2
⑫ 0
⑬ 0-1
⑭ 11-0-9-1
⑮ 12-11

Control Character Representations

ACK	Acknowledge
BEL	Bell
BS	Backspace
BYP	Bypass
CAN	Cancel
CC	Cursor Control
CR	Carriage Return
CU1	Customer Use 1
CU2	Customer Use 2
CU3	Customer Use 3
DC1	Device Control 1
DC2	Device Control 2
DC4	Device Control 4
DEL	Delete
DLE	Data Link Escape
DS	Digit Select
EM	End of Medium
ENQ	Enquiry
EOT	End of Transmission
ESC	Escape
ETB	End of Transmission Block
ETX	End of Text
FF	Form Feed
FS	Field Separator
HT	Horizontal Tab
IFS	Interchange File Separator
IGS	Interchange Group Separator
IL	Idle
IRS	Interchange Record Separator
IUS	Interchange Unit Separator
LC	Lower Case
LF	Line Feed
NAK	Negative Acknowledge
NL	New Line
NUL	Null
PF	Punch Off
PN	Punch On
RES	Restore
RS	Reader Stop
SI	Shift In
SM	Set Mode
SMM	Start of Manual Message
SO	Shift Out
SOH	Start of Heading
SOS	Start of Significance
SP	Space
STX	Start of Text
SUB	Substitute
SYN	Synchronous Idle
TM	Tape Mark
UC	Upper Case
VT	Vertical Tab

Special Graphic Characters

¢	Cent Sign
.	Period, Decimal Point
<	Less-than Sign
(	Left Parenthesis
+	Plus Sign
|	Logical OR
&	Ampersand
!	Exclamation Point
$	Dollar Sign
*	Asterisk
)	Right Parenthesis
;	Semicolon
¬	Logical NOT
-	Minus Sign, Hyphen
/	Slash
,	Comma
%	Percent
_	Underscore
>	Greater-than Sign
?	Question Mark
:	Colon
#	Number Sign
@	At Sign
'	Prime, Apostrophe
=	Equal Sign
"	Quotation Mark

AMERICAN NATIONAL STANDARDS INSTITUTE, INC. - COMMITTEE CORRESPONDENCE

Reply to: Eric H. Clamons
RD 1
North Wales, Pa. 19454
(215)962-5868

1970 August 3

Mr. V. E. Henriques
BEMA
1828 L St., N.W.
Washington, D. C. 20036

Dear Vico:

Pursuant to an ANSI directive and as unanimously approved by X3L2, please publicize in all trade journals and to all committees of X3 the following excerpt from document X3L2/964:

> "The American National Standard X3.4-1968 concerns itself with the code for information interchange. This code has been referenced as ASCII and later as USASCII. The policies prohibit the use of ANS as a prefix or part of an acronym. For the future, when referencing the standard code, the preferred acronym will be ASCII. USASCII, having been used, will be recognized. Future publications should use ASCII. USASCII should not be used in new documents. No one is requested to rework existing publications to effect the change. Gradual elimination of the USASCII acronym will be the desired goal. In no event should a new acronym starting with ANS gain any stature."

You may also wish to confirm and publicize the "ANSI Directive".

Very truly yours,

Eric H. Clamons
Chairman X3L2

EHC/jas

NMAH 311/2

X3L2/1017
Project #12
70-08-31
X3/SSC/133

AMERICAN NATIONAL STANDARDS INSTITUTE, INC. - COMMITTEE CORRESPONDENCE

Reply to: Eric H. Clamons
RD 1
North Wales, Pa. 19454
(215)962-5868

1970 August 31

On Delete Functions

A working paper prepared for SSC

Delete functions are concerned with the overt correction of text matter generated by a keyboard because the physical handling of the paper by erasure, opaquing, or retyping is not desirable or possible. The object of this paper is to prepare the technical facts necessary to determine the feasibility of standardizing the interface between human beings and the keyboard and also, of the corrective action desired.

The delete function is part of a broader class of text correction functions which also include "insert", "replace", "locate", etc.; these will not be discussed here because they are part of a specialized field. The delete functions discussed here are carried out during the text entry phase of a system and, as such, differ from generalized actions on stored text in that they operate only on the last character or the last string of characters entered in a data stream. They are immediate actions.

These delete funtions all have one element in common: The net effect of their action is that the resulting string of text must be unaware of the previous existence of the deleted matter or of the action to delete it, and the result will be digitized for computer processing and storage.

There are three input systems which can use delete functions: OCR systems in which text typed is intended for later reading in an OCR reader and it is considered too time consuming to opaque or obliterate the text manually; "machine aided" text entry systems which permit ad hoc corrections during text entry; and the traditional paper tape system which permits obliteration of the punched character. With all the commonality between them, the physical difference between them appears to have spawned different techniques for accomplishing the same thing for the three entry media and, worse, differences among themselves.

Delete Functions in OCR

Currently OCR Systems do not use Delete functions. Two special characters have been proposed:

NMAH 3" /6

- * Character Erase, a character which when printed over another or by itself will be ignored by the reader.
- * Group Erase, a character which when detected by the system will expunge a string of characters - usually a line.

Character Erase is simple to implement so that it never appears in the data stream. One of the characters proposed simply obliterates any character it overprints and even if it is read by itself can be identified as a character to be ignored. Combined with a "Repeat" action on a typewriter, it could be used to expunge a string of characters as well. A technique, which uses a series of wide dashes through text (in effect a continuous line through a string), can also be used for a repeated character erase. Unwanted text deleted by dashes is still readable, and the deletion can be accomplished equally well with a ruler during proofreading. The wide dash may not serve well as a single character delete symbol since it is primarily its continuity which detects the delete function.

Group Erase in which a single character erases a group of characters is more complicated because it presumes the presence of a memory, the ability to erase the memory, and to move the pointer in the memory back to the space of the last character just erased. The expediency for a typist to abandon a line must here be weighed against the complexity of implementation. The major problem is that whatever characters are chosen for delete function, they will decrease the repertoire of the set of OCR characters. The set is limited by the number of characters which can be generated from an office typewriter. This poses only a small problem in English speaking countries but a mounting problem in countries whose alphabet exceeds the basic 26 characters. OCR A and B both provide for "alphabetic extenders". Most office typewriters today have 44 keys, but some have only 43 while others have 45 and even 46. The most likely characters to be sacrificed in the US are ½ and ¼ since they cannot be read optically.

A final consideration for OCR input systems is the question of whether "type outs" and opaquing fluids wouldn't serve as well as the machine recognition techniques discussed above. Here the skill of the typist is a major factor. The higher the skill of the operator, the less significant the delete function becomes.

Delete Function in Machine Aided Text Entry Devices

Both character-delete and group-delete are used in many of the text entry systems in the field today. Some are used in time-sharing systems, some in text editing systems, and others in data preparation equipment. Some use typewriters, some use display devices (CRT's), and others use "blind" keyboards. The discussion of the delete functions is general and is not broken into equipment classes.

The character-delete function can be implemented as follows:

1. A delete character erases the last character of a string; repeated character-deletes each take out one character at a time.

   a) The delete character is visible. A visible character is useful to orient oneself in case of an interruption or lapse of memory. This system introduces additional characters on the typed line which makes tabular work more difficult.

   b) The delete character is not visible. Here, the printed line is not distorted as much as it is when the delete character is visible. It makes tabular work slightly easier to reconstruct. Since one cannot see whether delete was entered or how many times it requires more operator attention and leaves no record of the action taken, character-delete creates a null string which is expunged from the final character string. The function requires a memory and a pointer.

2. Backspace and Overprint
   Instead of using a delete character, one backspaces and overstrikes that position with a new character. This system is useful when it is necessary to know exactly where on the line one is while typing, e.g., when typing tables. Its drawback is the possibility of not knowing which character was struck and which one replaces it. A major drawback is that overstriking of one character by another, using backspace, is not possible. One needs two backspace characters, one with and one without character delete, and even then the printed record leaves doubt as to which action was taken.
3. Backspace and Erase
   The action is similar to the one above (2) but uses a feature which only display devices have. Unfortunately, display devices rarely permit the overprinting of characters. In any case, the drawback of poor readability of the result is overcome here.

4. Backspace and Obliterate
   This feature is used primarily in paper tape. The delete character over-punches the unwanted character and enters the data stream as a non-printing, non-spacing character.

   a) The function of the associated printer is similar to that of the non-visible character delete and still suffers from disturbing the tabular format of the printed line.

b) The function of the printer is to obliterate the character by overprinting (similar to OCR).

In paper tape this character may or may not disappear, depending on the system through which it would be processed. Generally, it is not advisable to eliminate the delete character in paper tape indiscriminately because it may have been inserted as a fill character. For instance, in some systems a backspace character requires a delete character to follow it for time fill.

The group-delete function can be implemented as follows:

1. A group-delete character can be visible and will delete a string of characters back to a designated delimiter.

    a) Usually it deletes a line. (similar to OCR)

    b) Another, fixed delimiting character is used, e.g. one of the ASCII Separators.

    c) A variable delimiting character is defined (see below, ASCII)

2. A group-delete character can be non-visible, having the same functions as the one above. If the character cannot be seen, the device usually displays visibly that the line was deleted, for instance, types in "delete".

In both visible and non-visible group delete, a carriage return is usually forced by the entry device (in OCR systems the operator does it). The advantage of an invisible group delete is that it does not reduce the character repertoire; it is probably preferred in devices capable of indicating that an action was taken. The visible group delete is closely related to the one discussed in OCR systems. Group deletes, which eliminate a line at a time, are probably the easiest to implement since most entry systems use the line as a base record. To delete a string back to a delimiting character requires more intelligence on the part of the apparatus.

Delete Functions in ASCII

ASCII provides a delete code which is defined strictly as a code used in paper tape to obliterate an undesirable punching. Generally the effect of the delete in paper tape is that it is ignored by the printing device, thus it gives the appearance of deletion on the printed copy while in fact it is not deleted from tape except by overt action on the part of an individual or a computer. Since the delete can also be used as a time fill or space fill character, it must not be destroyed in blind (transparent) interchange of data.

ASCII also provides a Cancel character which is not too well defined. Originally Cancel was a quasi-communications control character which deleted an entire message. A recent proposal from ECMA defines Cancel as the first character of a two-character sequence which causes the string preceding it to be deleted up to but not including the occurrence of the second character of the Cancel character sequence. The two-character Cancel sequence itself is also deleted. This proposal is more difficult to implement than the other group deletes since it requires a memory and search capability. The use of Cancel as a line delete function would more closely approach its original intent.

Since this is a first attempt at defining a possible systems problem, this paper does not pretend to be exhaustive. It is a summary of a discussion of SSC and should form the basis for further refinements and finally result in a statement of feasibility of standardization. Comments should be sent to the author in time for mail delivery on September 24, 1970.

Eric H. Clamons

EHC/jas

L2

NMAH 311/2 (Date?)

Subject: A consistent philosophy for deletion techniques

R. Bemer

In the course of developing time sharing systems, a number of conventions for deleting characters and lines has been developed. Regretably, many of these have been expedients. Some have overlooked the philosophy underlying the transmission code (ASCII). Others have employed terminals of limited character repertoires, forcing the reassignment of certain characters to these functions. Others have simply userpro certain characters arbitarily, because they were 'available' (meaning that they were not required in the languages immediately under consideration).

A method for deletion should meet the following criteria: It should be the same for all systems; It should be easy to remember and to employ it should not ascribe a new meaning of a character; It should be consistent with the philosophy of the code - in this case, ASCII.

A technique which is inconsistent particularly since it uses the character assigned in the code to perform this function, has been proposed by ECMA. It uses the CANCEL character followed by the character back to which, and including, characters are to be deleted. This allows to deletion of a single character, several characters, a repeated series of deletions (where the starting character appears several times on the string to be deleted, and of a line. To site some examples (CAN indicates the character CANCEL):

The deletion of a single character

Now is thr CAN e

produces

Now is the

Deletion of several characters

Now os the time CAN os is the
results

Now is the

Repeated deletion

Now os the time for CAN o CAN o is

results in

Now is

The effect of this first sequence of CAN o just after the 'for is:

Now os the time f CAN o is back to first preceding

The second sequence then operates back to the 'O' of 'os'; resulting in:

Now is

Deletion of a line (LF is LINE FEED):

Now is the time for all Good men to CAN LF good men
becomes

Now is the time for all good men

Deletion of a line, with run up

Deletion of a line, with run up

Now is the time for all good men to CAN CR good men to come

results in

Now is the time for all good men to come

Most current practice is based on the requirement that a graphick character be employed. The arbitrary selections which have been made have produced undesirable results when certain languages are later installed on the system. For example:

System	Delete character	conflict
IPC Time-Sharing		GE 600 ALGOL, APL
GE 600 TSS	@	I Language
Valley Forge	@ and	GE 600 ALGOL, I Language

The reason advanced for the use of the graphic character is that the user must see the result of his deletion.

However, this also demands that the user count quite carefully, to avoid erasing beyond the desirable point. For several characters, the cancel-to-character seems to possess greater merit, and seems to prevent no serious handicap even to single character deletion. Certainly it is for more flexible.

The cancel-to-character technique should be offered in all systems. If the user demands a graphic character, then the character should be selected by the user with a system command, with a system command, with the normal state of the system permitting the use of a complete character set of

ASCII. The change of the delete code at any time should be permitted (as is the meta character in TRAC).

Another system mode, which the user can invoke and revoke at will with a system command is one which causes the printing, at the end of that line, of any line in which a deletion has occurred.

Summary

The cancel-to-character is particularly attractive in its simplicity, versability, and in its potential for eliminating the conflicts in this practice among the various time sharing systems. An additional advantage is in data collection systems using bidirectional media. The circuitry required to do this (a single-character register) makes it extremely useful for error elimination during duplication, reducing requirements for media, and for transmission time in on-line systems.

SAYONARA
bye

R.W.B. MEB B-1..

NM Att 3 10/6

TC 1 -71/16A

EXTENDED BCD INTERCHANGE CODE LATIN ALPHABETS	3-3220 Subject	002 Suffix	CSS Cat.	70/11 Date

IBM Corporate Systems Standard

this is a third generation copy, reduced twice! not bad even for my bifocals?

Bob

INTRODUCTION

1.1 SCOPE. This standard defines for the IBM Corporation the BCD coded representation for up to 256 graphics and controls in punched cards, in magnetic tape, on data transmission lines, and in 8-bit BCD CPUs. It also defines a collating sequence.

This standard includes:

a) The 256 character Extended BCD Card Code.

b) The 256 character, 8-bit Extended BCD CPU Code.

c) The 256 character, 8-bit Extended BCD Magnetic Tape Code.

d) The 256 character, 8-bit Extended BCD Transmission Code.

e) The collating sequence for the 256 characters.

f) Control characters, graphic characters, and characters with special meaning.

g) Names for some of the characters.

h) Information on the 128-character ISO 7-Bit Code, on the 128-character American National Standard Code for Information Interchange, on the 128-character Hollerith Card Code, and the correspondences of those codes to 128 characters of the 256-character 8-bit Extended BCD Interchange Code.

i) Information on signed numerics.

j) Information on packed decimal.

k) Information on WTC graphic options.

l) Information on graphic options.

m) The 8-bit IBM 96-column card code correspondence to EBCDIC.

n) Information on MICR special symbols.

Compare with table IX p34
" XI p40

¢ ~ [
! ~]
| ~ !
¬ ~ ∧

conflict

condition for ASCII-1967 was that IBM change the lower two; these are homomorphic qua ASCII 1967

hich this document may be filed

TC 1 -71/16A

TABLE IX. EXTENDED BCD INTERCHANGE CODE

BITS 4 5 6 7	SECOND HEX DIGIT	DIGIT PUNCHES	00				01				10				11				BITS 0,1
			00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11	BITS 2,3
			0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	FIRST HEXADECIMAL DIGIT
			12				12	12		12	12	12		12	12				ZONE PUNCHES
				11				11	11	11		11	11	11		11			
					0		0		0	0	0		0	0			0		
			9	9	9	9	9	9	9	9									DIGIT PUNCHES
0000	0	8-1	① NUL	② DLE	③ DS	④	⑤ SP	⑥ &	⑦ -	⑧					⑨ {	⑩ }	⑪ \	⑫ 0	8-1
0001	1	1	SOH	DC1	SOS				⑬ /		a	j	~		A	J	⑭	1	1
0010	2	2	STX	DC2	FS	SYN					b	k	s		B	K	S	2	2
0011	3	3	ETX	TM							c	l	t		C	L	T	3	3
0100	4	4	PF	RES	BYP	PN					d	m	u		D	M	U	4	4
0101	5	5	HT	NL	LF	RS					e	n	v		E	N	V	5	5
0110	6	6	LC	BS	ETB	UC					f	o	w		F	O	W	6	6
0111	7	7	DEL	IL	ESC	EOT					g	p	x		G	P	X	7	7
1000	8	8	GE	CAN							h	q	y		H	Q	Y	8	8
1001	9	8-1	RLF	EM						`	i	r	z		I	R	Z	9	9
1010	A	8-2	SMM	CC	SM		¢	!	⑮ ¦	:								LVM	8-2
1011	B	8-3	VT	CU1	CU2	CU3	.	$	,	#									8-3
1100	C	8-4	FF	IFS		DC4	<	*	%	@					⑀		⑁		8-4
1101	D	8-5	CR	IGS	ENQ	NAK	(	)	_	'									8-5
1110	E	8-6	SO	IRS	ACK		+	;	>	=					⑂				8-6
1111	F	8-7	SI	IUS	BEL	SUB	|	¬	?	"								EO	8-7
			12				12				12	12		12	12	12		12	Zone Punches
				11				11				11	11	11		11	11	11	
					0				0		0		0	0	0		0	0	
			9	9	9	9									9	9	9	9	

Card Hole Patterns

① 12-0-9-8-1
② 12-11-9-8-1
③ 11-0-9-8-1
④ 12-11-0-9-8-1
⑤ No Punches
⑥ 12
⑦ 11
⑧ 12-11-0
⑨ 12-0
⑩ 11-0
⑪ 0-8-2
⑫ 0
⑬ 0-1
⑭ 11-0-9-1
⑮ 12-11

TC 1 -71/16A

CUSTOMER USE CHARACTERS

REF. #	GRAPHICS	NAME	DESCRIPTION SEE SECTION
27	CU1	Customer Use 1	2.4.2.27
43	CU2	Customer Use 2	2.4.2.43
59	CU3	Customer Use 3	2.4.2.59

DEVICE CONTROL CHARACTERS

REF. #	GRAPHICS	NAME	DESCRIPTION SEE SECTION
4	PF	Punch Off	2.4.2.4
17	DC1	Device Control 1	2.4.2.17
18	DC2	Device Control 2	2.4.2.18
19	DC3/TM	Device Control 3/ Tape Mark	2.4.2.19
52	PN	Punch On	2.4.2.52
53	RS	Reader Stop	2.4.2.53
60	DC4	Device Control 4	2.4.2.60

ERROR CONTROL CHARACTERS

REF. #	GRAPHICS	NAME	DESCRIPTION SEE SECTION
7	DEL	Delete	2.4.2.7
24	CAN	Cancel	2.4.2.24
45	ENQ	Enquiry	2.4.2.45
46	ACK	Acknowledge	2.4.2.46
47	BEL	Bell	2.4.2.47
61	NAK	Negative Acknowledge	2.4.2.61
63	SUB/CH	Substitute Check	2.4.2.63
64	EO	Eight Ones	2.4.2.64

FORMATTING OR EDITING CONTROL CHARACTERS

REF. #	GRAPHICS	NAME	DESCRIPTION SEE SECTION
5	HT	Horizontal Tab	2.4.2.5
9	RLF	Reverse Line Feed	2.4.2.9
11	VT	Vertical Tab	2.4.2.11
12	FF	Form Feed	2.4.2.12
13	CR	Carriage Return	2.4.2.13
21	NL	New Line	2.4.2.21
22	BS	Back Space	2.4.2.22
32	DS	Digit Select	2.4.2.32
33	SOS	Start of Significance	2.4.2.33
34	FS	Field Separator	2.4.2.34
37	LF	Line Feed	2.4.2.37

EXTENDED BCD INTERCHANGE CODE
LATIN ALPHABETS

TABLE XI. EBCDIC AND 7-BIT CODE/HOLLERITH CORRESPONDENCES

BITS 4 5 6 7	SECOND HEX DIGIT	DIGIT PUNCHES	00				01				10				11				BITS 0,1
			00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11	BITS 2,3
			0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	FIRST HEX DIGIT
			12				12	12		12	12	12		12	12				ZONE PUNCHES
				11				11	11	11		11	11	11		11			
					0		0		0	0	0		0	0			0		
			9	9	9	9	9	9	9	9									DIGIT PUNCHES
0000	0	8-1	① NUL 0/0	② DLE 1/0	③	④	⑤ SP 2/0	⑥ & 2/6	⑦ - 2/13	⑧					⑨ { 7/11	⑩ } 7/13	⑪ \ 5/12	⑫ 0 3/0	8-1
0001	1	1	SOH 0/1	DC1 1/1					⑬ / 2/15		a 6/1	j 6/10	~ 7/14		A 4/1	J 4/10	⑭	1 3/1	1
0010	2	2	STX 0/2	DC2 1/2		SYN 1/6					b 6/2	k 6/11	s 7/3		B 4/2	K 4/11	S 5/3	2 3/2	2
0011	3	3	ETX 0/3	DC3 1/3							c 6/3	l 6/12	t 7/4		C 4/3	L 4/12	T 5/4	3 3/3	3
0100	4	4									d 6/4	m 6/13	u 7/5		D 4/4	M 4/13	U 5/5	4 3/4	4
0101	5	5	HT 0/9		LF 0/10						e 6/5	n 6/14	v 7/6		E 4/5	N 4/14	V 5/6	5 3/5	5
0110	6	6		BS 0/8	ETB 1/7						f 6/6	o 6/15	w 7/7		F 4/6	O 4/15	W 5/7	6 3/6	6
0111	7	7	DEL 7/15		ESC 1/11	EOT 0/4					g 6/7	p 7/0	x 7/8		G 4/7	P 5/0	X 5/8	7 3/7	7
1000	8	8		CAN 1/8							h 6/8	q 7/1	y 7/9		H 4/8	Q 5/1	Y 5/9	8 3/8	8
1001	9	8-1		EM 1/9						` 6/0	i 6/9	r 7/2	z 7/10		I 4/9	R 5/2	Z 5/10	9 3/9	9
1010	A	8-2					[5/11	] 5/13	⑮ 7/12	: 3/10									8-2
1011	B	8-3	VT 0/11				. 2/14	$ 2/4	, 2/12	# 2/3									8-3
1100	C	8-4	FF 0/12	FS 1/12		DC4 1/4	< 3/12	* 2/10	% 2/5	@ 4/0									8-4
1101	D	8-5	CR 0/13	GS 1/13	ENQ 0/5	NAK 1/5	(2/8	) 2/9	_ 5/15	' 2/7									8-5
1110	E	8-6	SO 0/14	RS 1/14	ACK 0/6		+ 2/11	; 3/11	> 3/14	= 3/13									8-6
1111	F	8-7	SI 0/15	US 1/15	BEL 0/7	SUB 1/10	| ! 2/1	¬ ^ 5/14	? 3/15	" 2/2									8-7
			12				12				12	12		12	12	12		12	Zone Punches
				11				11				11	11	11		11	11	11	
					0				0		0		0	0	0		0	0	
			9	9	9	9									9	9	9	9	

Card Hole Patterns

① 12-0-9-8-1	⑤ No Punches	⑨ 12-0	⑬ 0-1
② 12-11-9-8-1	⑥ 12	⑩ 11-0	⑭ 11-0-9-1
③ 11-0-9-8-1	⑦ 11	⑪ 0-8-2	⑮ 12-11
④ 12-11-0-9-8-1	⑧ 12-11-0	⑫ 0	

relation to 8-bit
ASCII not
in their Std.
*probably waiting for
X3.26-1970

IBM

TECHNICAL
BULLETIN

THE ASCII CODES
(1963, 1967 and 1968 Versions)

western union

August 1971

THE AMERICAN NATIONAL STANDARD CODE FOR INFORMATION INTERCHANGE (ASCII), X3.4-1963

b7 →				0	0	0	0	1	1	1	1
b6 →				0	0	1	1	0	0	1	1
b5 →				0	1	0	1	0	1	0	1
b4 ↓	b3 ↓	b2 ↓	b1 ↓								
0	0	0	0	NULL	DC$_0$	ƀ	0	@	P	UNASSIGNED	UNASSIGNED
0	0	0	1	SOM	DC$_1$	!	1	A	Q		
0	0	1	0	EOA	DC$_2$	"	2	B	R		
0	0	1	1	EOM	DC$_3$	#	3	C	S		
0	1	0	0	EOT	DC4 STOP	$	4	D	T		
0	1	0	1	WRU	ERR	%	5	E	U		
0	1	1	0	RU	SYNC	&	6	F	V		
0	1	1	1	BELL	LEM	' (APOS)	7	G	W		
1	0	0	0	FE$_0$	S$_0$	(	8	H	X		
1	0	0	1	HT / SK	S$_1$	)	9	I	Y		
1	0	1	0	LF	S$_2$	*	:	J	Z		
1	0	1	1	VTAB	S$_3$	+	;	K	[		
1	1	0	0	FF	S$_4$	(COMMA) ,	<	L	\		ACK
1	1	0	1	CR	S$_5$	(HYPHEN) -	=	M	]		①
1	1	1	0	SO	S$_6$	(PERIOD) .	>	N	↑		ESC
1	1	1	1	SI	S$_7$	/	?	O	←		DEL

THE AMERICAN NATIONAL STANDARD CODE FOR INFORMATION INTERCHANGE (ASCII) X3.4-1968

	b7 →					0	0	0	0	1	1	1	1
	b6 →					0	0	1	1	0	0	1	1
	b5 →					0	1	0	1	0	1	0	1
	b4 ↓	b3 ↓	b2 ↓	b1 ↓	COL → ROW ↓	0	1	2	3	4	5	6	7
	0	0	0	0	0	NUL	DLE	SP	0	@	P	`	p
	0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
	0	0	1	0	2	STX	DC2	"	2	B	R	b	r
	0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
	0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
	0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
	0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
	0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
	1	0	0	0	8	BS	CAN	(	8	H	X	h	x
	1	0	0	1	9	HT	EM	)	9	I	Y	i	y
A	1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
B	1	0	1	1	11	VT	ESC	+	;	K	[	k	{
C	1	1	0	0	12	FF	FS	,	<	L	\	l	¦
D	1	1	0	1	13	CR	GS	-	=	M	]	m	}
E	1	1	1	0	14	SO	RS	.	>	N	^	n	~
F	1	1	1	1	15	SI	US	/	?	O	_	o	DEL

NULL	–	Null/Idle
SOM	–	Start of Message (CC)
EOA	–	End of Address (CC)
DC$_0$	–	Device control reserved for Data Link Escape
DC$_1$	–	Device Control 1
DC$_2$	–	Device Control 2
DC$_3$	–	Device Control 3
DC4 (STOP)	–	Device Control 4 (Stop)
ERR	–	Error (CC)
EOM	–	End of Message (CC)
EOT	–	End of Transmission (CC)
WRU	–	"Who are you?" (CC)
RU	–	"Are you. . .?" (CC)
BELL	–	Audible Signal
FE$_0$	–	Format Effector (FE)
HT	–	Horizontal Tabulation (FE)
SK	–	Skip (punched card) (FE)
LF	–	Line Feed (FE)
VTAB	–	Vertical Tabulation (FE)
FF	–	Form Feed (FE)
CR	–	Carriage Return (FE)
SO	–	Shift Out
SI	–	Shift In
SYNC	–	Synchronous Idle (CC)
LEM	–	Logical End of Media (CC)
S$_0$ - S$_7$	–	Separator (information)
ƀ	–	Space (Word separator, normally non-printing)
<	–	Less Than
>	–	Greater Than
↑	–	Up Arrow (Exponentiation)
←	–	Left Arrow (Implies/Replaced by)
\	–	Reverse Slant
ACK	–	Acknowledge (CC)
①	–	Unassigned Control
ESC	–	Escape
DEL	–	Delete/Idle

NOTE:

(CC) = **Communication Control**
(FE) = **Format Effector**
(IS) = **Information Separator**

NUL	–	Null
SOH	–	Start of Heading (CC)
STX	–	Start of Text (CC)
ETX	–	End of Text (CC)
EOT	–	End of Transmission (CC)
ENQ	–	Enquiry (CC)
ACK	–	Acknowledge (CC)
BEL	–	Bell (audible or attention signal)
BS	–	Backspace (FE)
HT	–	Horizontal Tabulation (punched card skip) (FE)
LF	–	Line Feed (FE)
VT	–	Vertical Tabulation (FE)
FF	–	Form Feed (FE)
CR	–	Carriage Return (FE)
SO	–	Shift Out
SI	–	Shift In
DLE	–	Data Link Escape (CC)
DC1	–	Device Control 1
DC2	–	Device Control 2
DC3	–	Device Control 3
DC4	–	Device Control 4 (stop)
NAK	–	Negative Acknowledge (CC)
SYN	–	Synchronous Idle (CC)
ETB	–	End of Transmission Block (CC)
CAN	–	Cancel
EM	–	End of Medium
SUB	–	Substitute
ESC	–	Escape
FS	–	File Separator (IS)
GS	–	Group Separator (IS)
RS	–	Record Separator (IS)
US	–	Unit Separator (IS)
SP	–	Space
DEL	–	Delete

NOTE:

(CC) = **Communication Control**
(FE) = **Format Effector**
(IS) = **Information Separator**

CBI 94/2/ unnumbered

HISTORY OF ASCII

On June 17, 1963 a new American Standard Code for Information Interchange was approved by what was then the American Standards Association (ASA). The new code, commonly referred to as ASCII (pronounced as-key), is a 7-bit code intended for information interchange among data processing systems, communications systems, and associated equipments. Twenty-eight of the 128 possible bit permutations were unassigned in the original version of the code. However, in 1967 The USA Standards Institute, successor to the American Standards Association*, revised the code by assigning characters to all previously unassigned bit permutations and by making some changes in previously assigned characters. In 1968 minor changes were made in the code, and a revised version of the USA Standard Code for Information Interchange, X3.4-1968, was issued. This technical bulletin explains the differences between the three versions of the code and describes how the code is implemented in Western Union's 8-level ASCII code equipment.

[illegible]

In discussing the 1963 ASCII code, it is convenient to arrange the 128 bit permutations in 8 vertical columns, with 16 horizontal rows in each column, as shown in the 1963 ASCII table on the cover page. The vertical columns are identified by the three high-order bits (b7, b6, and b5) and the 16 horizontal rows by the four low-order bits (b4, b3, b2, and b1). The bit pattern for any character is usually given in the high-order to low-order bit sequence. Thus the bit pattern or code for the letter R is:

Bit number:	7	6	5	4	3	2	1
Bit pattern:	1	0	1	0	0	1	0

The first two columns of the code table contain control characters, which are normally non-printing characters. The next four columns contain 64 graphic characters. The last two columns are unassigned, except for four control characters in the last column. (Strictly speaking, the delete character in the bottom row of the last column is not a control character. It is used for deleting errors in punched paper tape by overpunching the errored characters.)

CLASSES OF CONTROL CHARACTERS

The control characters in the ASCII 63 code fall into four primary classes: communications controls; format effectors; device controls; and information separators. Communications controls such as SOM (Start of Message) and EOA (End of Address), are used primarily in communications. Format effectors such as carriage return (CR) and line feed (LF), are used to organize printed data on a page. Device controls are used to control operation of auxiliary equipment. (Use of the device controls has not been standardized, but they are used in some Western Union Systems for turning an auxiliary reperforator on or off and for starting or stopping a tape transmitter.) Information separators are used to identify boundaries of various elements of information. These separators, also called data delimiters, are machine-oriented controls used in data processing to serve a purpose similar to that of punctuation marks in message communications.

Several special controls which do not logically fall into one of the four primary classes previously described are also included in the code. These special controls are not normally used in Western Union systems. They may be used in some systems which use ASCII-code terminals to communicate with a computer.

[illegible]

In 1967 the ASCII code was revised to include the lower case of the alphabet, and to incorporate other changes to make the code more useful for both communications and data processing.

COLUMN/ROW DESIGNATIONS

In the revised code, illustrated on the cover page, the eight columns are numbered 0 through 7, from left to right. The column number is the decimal equivalent of the binary number formed by the three high order bits (b7, b6, b5). Similarly, the rows are numbered 0 through 15 from top to bottom, with the row number being the decimal equivalent of the binary number formed by the four low order bits (b4, b3, b2, b1). Any character in the code may, therefore, be represented by giving its column and row position in the code table. For example, the capital letter G may be represented by the notation "column 4, row 7," or simply by 4/7.

GRAPHIC SUBSET

The 26 lower case letters were placed in columns 6 and 7, and arranged in increasing binary sequence from a to z. Each lower case letter was placed in the same row as the corresponding upper case letter. This arrangement results in a simple one-bit difference between upper and lower case letters. Dual case keyboards for generating the code can, therefore, be designed to generate upper case letters by using a shift key to invert bit 6. The non-alpha characters were placed in column 7 to obtain logical pairings with corresponding characters in column 5 (e.g., deleting or inverting bit 6 will change the bit permutation for the opening brace to that for the opening bracket).

---

*Effective October 6, 1969 the name of the Institute was changed to American National Standards Institute, Inc.

CONTROL CHARACTERS

In the 1967 ASCII, three control characters were relocated. The desirability in data processing of being able to readily identify different classes of graphics and controls, coupled with the likelihood of a "hit," or short open, on a communication line generating a false control in column seven, prompted this change. The acknowledge (ACK) and escape (ESC) characters were placed in the first two columns. The unassigned control in column 7, row 13 was dropped. In order to accommodate these controls in the first two columns, four of the eight information separators and the "are you" (RU) character included in the original code were dropped from the code table. The four remaining information separators were assigned the specific functions of file separator, group separator, record separator, and unit separator. These four characters are contiguous with SP (space), which can also be used as an information separator.

1968 ASCII

In 1968 minor changes were made in the text of the 1967 standard, but the code table remained unchanged. The 1968 revision permits the use of the notation ASCII or USASCII as proper identification of the standard and states that the notation should be taken to mean the latest issue. To designate a prior or particular issue of the standard, the last two digits of the year of issue may be affixed, as ASCII 63 or USASCII 67. The new revision also permits use of a single character to perform the combined line feed (LF) and carriage return (CR) functions. When a single character is used to perform both functions, the LF character shall be used and shall be designated New Line (NL).

The accompanying ASCII comparison chart reflects the differences between the original ASCII 63 and the revised ASCII 67/68. The lightly shaded areas illustrate minor differences in the two versions of the code; the heavily shaded areas identify major differences; the intermediately shaded areas identify new assignments to the original ASCII 63. Where only one character is listed and the area is not shaded, the code has remained the same. The accompanying comparison table lists character differences between ASCII 63 and ASCII 67/68 and includes an explanation of each difference.

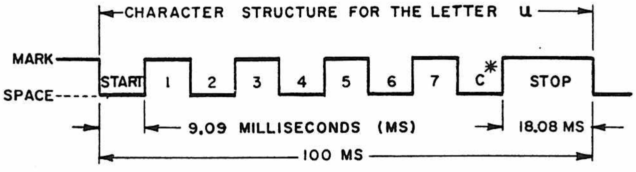
COMPARISON CHART, ASCII 63 VS. ASCII 68

b7 →					0		0		0		0		1		1		1		1	
b6 →					0		0		1		1		0		0		1		1	
b5 →					0		1		0		1		0		1		0		1	
b4 ↓	b3 ↓	b2 ↓	b1 ↓	COL →	0		1		2		3		4		5		6		7	
				ROW ↓	63	68	63	68	63	68	63	68	63	68	63	68	63	68	63	68
0	0	0	0	0	NULL	NUL	DCO	DLE	␢	SP	0		@		P		↑ UNASSIGNED ↓	\`	↑ UNASSIGNED ↓	p
0	0	0	1	1	SOM	SOH	DC1		!		1		A		Q			a		q
0	0	1	0	2	EOA	STX	DC2		"		2		B		R			b		r
0	0	1	1	3	EOM	ETX	DC3		#		3		C		S			c		s
0	1	0	0	4	EOT		DC4 STOP	DC4	$		4		D		T			d		t
0	1	0	1	5	WRU	ENQ	ERR	NAK	%		5		E		U			e		u
0	1	1	0	6	RU	ACK	SYNC	SYN	&		6		F		V			f		v
0	1	1	1	7	BELL	BEL	LEM	ETB	'		7		G		W			g		w
1	0	0	0	8	FE$_0$	BS	S$_0$	CAN	(		8		H		X			h		x
1	0	0	1	9	HT/SK	HT	S$_1$	EM	)		9		I		Y			i		y
1	0	1	0	10	LF		S$_2$	SUB	*		:		J		Z			j		z
1	0	1	1	11	VTAB	VT	S$_3$	ESC	+		;		K		[			k		{
1	1	0	0	12	FF		S$_4$	FS	,		<		L		\			l	ACK	¦
1	1	0	1	13	CR		S$_5$	GS	-		=		M		]			m	①	}
1	1	1	0	14	SO		S$_6$	RS	.		>		N		↑	^		n	ESC	~
1	1	1	1	15	SI		S$_7$	US	/		?		O		←	—		o	DEL	

COMPARISON TABLE, ASCII 63 VS. ASCII 68

COL/ROW IN CODE TABLE	1963 VERSION	1968 VERSION	EXPLANATION OF CHANGE
0/0	NULL	NUL	Mnemonic Shortened
0/1	SOM	SOH	Start of Message changed to Start of Heading
0/2	EOA	STX	End of Address changed to Start of Text
0/3	EOM	ETX	End of Message changed to End of Text
0/5	WRU	ENQ	Who Are You changed to Enquiry
0/6	RU	ACK	RU replaced by Acknowledge, moved from position 7/12 to 0/6
0/7	BELL	BEL	Mnemonic shortened
0/8	FE_0	BS	Back space assigned to previously unassigned format effector
1/0	DC_0	DLE	Data Link Escape assigned to previously unassigned Device Control
1/5	ERR	NAK	Error changed to Negative Acknowledge
1/6	SYNC	SYN	Mnemonic Shortened
1/7	LEM	ETB	Logical End of Media replaced by End of Transmission Block. LEM changed to End of Media (EM) and moved to position 1/9
1/8	S_0	CAN	Information Separator replaced by Cancel
1/9	S_1	EM	Information Separator replaced by End of Media (EM was Logical End of Media)
1/10	S_2	SUB	Information Separator replaced by Substitute
1/11	S_3	ESC	Information Separator replaced by Escape. (ESC was in position 7/14)
1/12	S_4	FS	Information Separator assigned to File Separator
1/13	S_5	GS	Information Separator assigned to Group Separator
1/14	S_6	RS	Information Separator assigned to Record Separator
1/15	S_7	US	Information Separator assigned to Unit Separator
5/14	↑	^	Up arrow changed to Circumflex
5/15	←	—	Left arrow changed to underline
6/0	Unassigned	ˊ	Assigned to grave accent
6/1 thru 6/15	Unassigned	a thru o	Unassigned positions assigned to lower case alphabet
7/0 thru 7/10	Unassigned	p thru z	Unassigned positions assigned to lower case alphabet
7/11	Unassigned	{	Assigned to left or opening brace
7/12	ACK	¦	Acknowledge moved to position 0/6. Vertical line assigned to Position 7/12
7/13	Unassigned Control	}	Assigned to right or closing brace
7/14	ESC	~	Escape moved to 1/11. Overline assigned to position 7/14

The Models 33 and 35 teleprinters were originally designed for use with the 7-bit ASCII 63. The manufacturer chose to use an 8-bit code for implementing ASCII, employing the eighth bit as an even vertical parity check bit. (When the parity check bit is used, the keyboard will generate even parity when signals are transmitted or tape is punched from the keyboard, but parity will not be checked on incoming signals.) When even parity is not desired, the eighth bit can be made always marking. The manufacturer also chose to use a unit-length start pulse and a two-unit-length rest (stop) pulse, thus giving a modulation rate of 110 bauds (bits per second) at a speed of 100 words per minute. The ASCII bits are transmitted low order first. The following drawing illustrates the character structure for the capital letter U in the TWX ASCII code.



*C = Parity Check Bit

Certain characteristics of ASCII 63 permit simplified design of the printing mechanism for implementing the code in monocase Model 33 and 35 teleprinters. The 64-character graphic subset in the code is located in the middle four columns of the chart. In each of these columns bit 7 differs from bit 6. To determine whether a bit permutation is a printing character or control character, it is only necessary to compare these two bits. If bits 7 and 6 are both zeros or if both bits are ones, the character is not a graphic. If either is a one bit, and the other a zero bit, the character received is a graphic. In the Models 33 and 35 typing units, once a bit permutation has been identified as a graphic character, bit six is ignored; i.e., bit 6 is not used to position the typewheel (Model 33) or type box (Model 35). It should be noted that there are four discrete permutations (00, 01, 10, and 11) of bits 7 and 5 in the four graphic columns of the table, so that these two bits can be used to determine which column a graphic character is in, once it has been determined that a character is a graphic.

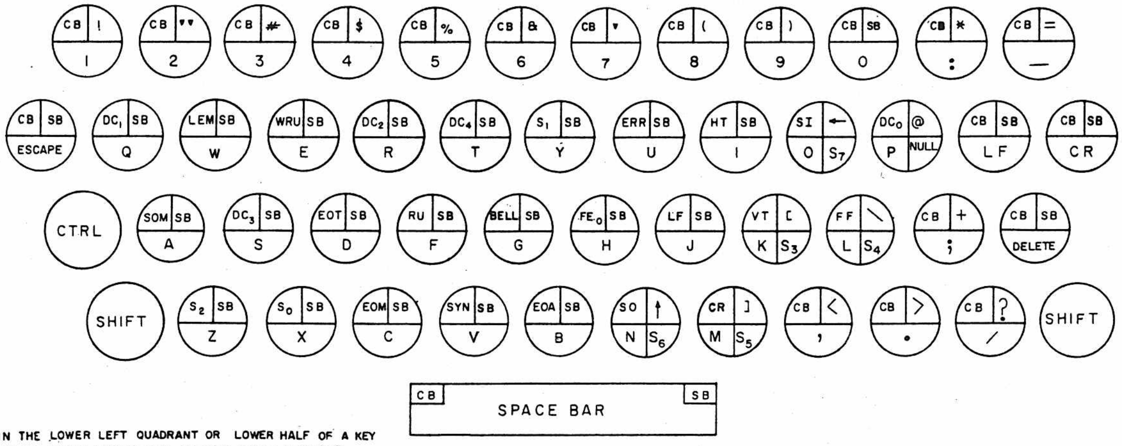
In the Models 33 and 35 ASCII sets, printing is suppressed when a control character in one of the first two columns of the code table is received. Printing is also suppressed in the

1963 ASCII sets when one of the three control characters or the DEL (delete) character in the last column is received.

GENERATION OF CONTROL CHARACTERS

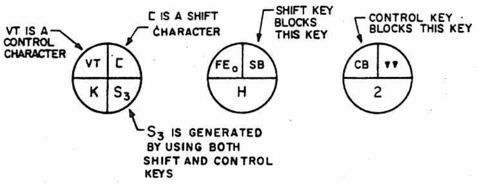
Most of the ASCII 63 control characters are located in the first two columns of the code table. In both of these columns, bit 7 is a zero. In the Models 33 and 35 keyboards, most of the controls in these first two columns are generated by means of a control key which, when held depressed, deletes bit 7; that is, if bit 7 is a one, the control key converts it to a zero. For example, to generate the start-of-message (SOM) character, the control key is held down while the "A" key is being depressed. All of the control characters in the first two columns of the code table can be generated, even though most of these characters are not shown on the keytops. The number of control characters actually imprinted on the keytops varies, depending on the system in which the set is used. The carriage return (CR), line feed (LF), escape (ESC), and delete (DEL) characters are generated without using the control key, since separate keys are provided for these three controls. (On some keyboards, the escape key is labeled ALT MODE and the delete key is labeled RUB OUT.) The following illustration describes how the remaining control characters and shift characters are generated from a keyboard.

HOW ASCII SHIFT AND CONTROL CHARACTERS ARE GENERATED



NOTES:

1. A CHARACTER SHOWN IN THE LOWER LEFT QUADRANT OR LOWER HALF OF A KEY TOP IS GENERATED WITHOUT USING EITHER THE CONTROL (CTRL) OR THE SHIFT KEY.
2. A CHARACTER SHOWN IN THE UPPER LEFT QUADRANT OF A KEYTOP IS GENERATED IN CONJUNCTION WITH THE CONTROL KEY.
3. A CHARACTER SHOWN IN THE UPPER RIGHT QUADRANT OF A KEYTOP IS GENERATED IN CONJUNCTION WITH THE SHIFT KEY.
4. A CHARACTER SHOWN IN THE LOWER RIGHT QUADRANT OF A KEYTOP IS GENERATED IN CONJUNCTION WITH BOTH CONTROL AND SHIFT KEYS.
5. CB IN THE UPPER LEFT QUADRANT OF A KEYTOP MEANS THAT THIS KEY IS BLOCKED BY THE CONTROL KEY.
6. SB IN THE UPPER RIGHT QUADRANT OF A KEYTOP MEANS THAT THIS KEY IS BLOCKED BY THE SHIFT KEY.
7. ON THE MODEL 35, THE CONTROL KEY ALSO BLOCKS THE "J" KEY. ON MODEL 33 IT DOES NOT.
8. EXAMPLE:


VT IS A CONTROL CHARACTER
[IS A SHIFT CHARACTER
SHIFT KEY BLOCKS THIS KEY
CONTROL KEY BLOCKS THIS KEY
VT [K S3
FE0 SB H
CB " 2
S3 IS GENERATED BY USING BOTH SHIFT AND CONTROL KEYS

In most Western Union systems, Device Control 1 (DC1) in the code is used as a transmitter-on control character and Device Control 3 (DC3) is used as a transmitter-off control. These two controls, abbreviated X-ON and X-OFF, are shown on the upper halves of the Q and S keytops, respectively, when they are used in a system. When the X-ON character is transmitted, it starts the transmitter at a distant station when the distant station is equipped to respond to X-ON. When the X-OFF character is punched in tape and read by a local tape reader, it stops the local reader if the local station is equipped to respond to X-OFF.

Similarly, Device Control 2 (DC2) in the code is used in some systems to automatically turn on a distant tape punch and Device Control 4 (DC4) is used to automatically turn off a distant tape punch, provided the distant station is equipped with the optional automatic punch-on and punch-off feature. These two controls, labeled TAPE and ~~TAPE~~, appear on the upper halves of the R and T keytops, respectively, when they are used in a system.

At present the Escape (ESC) character is not used in Western Union systems. However, it may be used in systems where a Model 33 or 35 is used to communicate with a computer.

The following table summarizes the control code designations used on Models 33 and 35 keyboards which are different from the designations used in the code table:

ASCII CONTROL CHARACTER KEYBOARD DESIGNATIONS USED ON MODELS 33 and 35

ASCII CONTROL	KEYBOARD DESIGNATION
FF (Form Feed)	FORM (Some sets)
CR (Carriage Return)	RETURN
DC1 (Device Control 1)	X-ON
DC2 (Device Control 2)	TAPE
DC3 (Device Control 3)	X-OFF
DC4 (Device Control 4)	~~TAPE~~
ESC (Escape)	ALT MODE (Some sets)
DEL (Delete)	RUB OUT

NOTE: Control character designations not shown are the same as the ASCII designations.

ASCII 68 TELEPRINTERS

The Models 33 and 35 sets which have been modified to operate on the ASCII 68 code incorporate changes in the typing unit and the keyboard. The typing unit changes are all a result of the changes made in the ASCII graphic characters and involve changes in the type box (Model 35) or typewheel (Model 33). The keyboard changes all involve changes in the keytop designations and are the result of changes made in both the graphic and control characters.

ASCII 68 TYPING UNITS

Two major changes were incorporated in the type wheels and type boxes of the ASCII 68 sets. The left arrow (←) was replaced by the underline (_) and the up arrow (↑) was replaced by the circumflex (^). Three other minor changes were also made in the typing units in order to enable them to conform more closely to the ASCII 68 and the ISO (International Standards Organization) 7-bit code. The ASCII 68 permits alternate use of three punctuation marks as diacritical marks; i.e., marks placed over or under a letter to show pronunciation. The quotation marks were modified to permit them to be used alternatively as diereses; the apostrophe was modified to permit it to be used alternatively as an acute accent or closing single quotation mark; and the comma was modified to permit it to be used alternatively as a cedilla.

The Models 33 and 35 modified for use with the ASCII 68 code suppress printing when a control character in column 0 or 1 is received and also suppress printing when the DEL character is received. Printing is not suppressed when the characters in positions 7/12, 7/13, and 7/14 are received.

In addition to the foregoing changes, two other changes were made in the ASCII 68 type fonts which were not related to the revision of ASCII. The letter O was made fatter and the zero was made slimmer to provide greater distinction between these two very similar characters.

The typing units of the Model 33 ASCII 63 sets are not parity sensitive; that is, the graphic characters will print correctly and the control characters will perform their functions correctly, regardless of whether the 8th (parity) bit is marking or spacing. However, the Model 33 ASCII 68 sets are sensitive to parity on some control characters, such as EOT and X-ON. The graphic characters and some controls, such as carriage return and line feed, are parity insensitive on the ASCII 68 sets.

If an ASCII 68 Model 33 set transmits to an ASCII 63 Model 33 there will be no problem due to failure of the latter set to respond to the control characters. However, if an ASCII 63 set arranged to transmit an 8th bit which is always marking transmits to an ASCII 68 set, some control characters, such as X-ON, may not operate the assigned function. If an ASCII 63 set sends even parity, this problem will not arise.

All typing units on Model 35 sets, whether the ASCII 63 or 68 version, are parity insensitive and control characters will perform their proper functions regardless of whether bit 8 is marking or spacing.

ASCII 68 KEYBOARDS

The keyboard changes in the ASCII 68 sets are primarily changes in keytop designations. The ASCII 68 sets, like the ASCII 63 sets, can generate all of the control characters in the first two columns of the code table. However, like the ASCII 63 sets, only those controls actually used in a system are normally imprinted on keytops. In the ASCII 63 teleprinters three graphics, the opening bracket ([ ), closing bracket ( ]), and reverse slash (\ ), were omitted. In the ASCII 68 sets these three characters are imprinted on the K, M, and L keytops, respectively. However, addition of these three graphics is not related to the revision of the code. The circumflex replaced the up arrow on the N key and the underline replaced the left arrow on the O key.

The start of heading (SOH), start of text (STX), and end of text (ETX) characters were added to the keytops of the ASCII 68 sets. In addition, the word FORM for Form Feed, on the L key, was changed to FF. The RU character, which appears on some ASCII 63 Model 33 sets, was dropped. These differences between the ASCII 63 and 68 sets are listed in the accompanying table. In addition to these changes, ASCII 68 keyboards which are equipped with an ESC (escape) key will generate the ASCII 68 code combination when depressed.

MODELS 33 AND 35 KEYBOARD CHANGES

ASCII 63 KEYTOP	ASCII 68 KEYTOP
A	A SOH
B	B STX
C	C ETX
F RU (Some Model 33's)	F
*K VT	K [VT
*L FORM	L \FF
*M	M]
N ↑	N ^
O ←	O _

*These changes were not made as a result of the changes in the ASCII code, but were incorporated in the keyboards at the same time as the ASCII 63 to 68 changes.

REFERENCE DOCUMENTATION

Model 33 TWX KSR and ASR Sets, Product Data Sheet 1-3.
Model 35 TWX ASR Sets, Product Data Sheet 1-4.
Model 35 TWX KSR and RO Sets, Product Data Sheet 1-5.
Model 35 ASR Set, Class 1 Terminal, Info-Com System, Product Data Sheet 1-6.
The USA Standard Code for Information Interchange, X3.4-1968.
(Available from the American National Standards Institute, 1430 Broadway, New York, N.Y. 10018)

Association Française
de Normalisation
Tour EUROPE
CEDEX 7
92 - PARIS-LA DEFENSE
Tél. PARIS (1) 788.11.11

I S O
INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
ORGANISATION INTERNATIONALE DE NORMALISATION

ISO/TC 97/SC 2
CHARACTER SETS AND CODING
Secretariat
AFNOR (France)

ISO/TC 97/SC 2
(Sec. 165)
481
November 1970

BRIEF REPORT ON THE MEETING OF
CCITT/GM - ALP
18 - 20 November 1970

9/113

Brief Report on the Meeting of
CCITT/GM-ALP

18 - 20 November 1970

Introduction:

The Joint Working Party on the "Use of alphabet N° 5" (correspond ing to ISO 7-bit coded character set) is of concern of the follow ing Study Groups of the CCITT:

COM I	:	Telegraph operation and tariffs (including telex)
COM VIII	:	Telegraph equipment and local connecting lines
COM X	:	Telegraph Switching
COM Sp.A	:	Data Transmission

The meeting was held in Geneva at the UIT headquarter, under the chairmanship of Mr. G. Baggenstos (Switzerland).

The agenda, reported in the following, is based on a question which is common to Study Group I, VIII and Sp.A, i.e. Question 6/I, a part of Question 9/VIII and Question 1/A point C.

Agenda:

1. Possible modification to Alphabet N°.5
   - 1.1 Note 7 in positions 2/3 and 2/4 of the code table
   - 1.2 Additional preferred graphic characters for "National Use" position
2. National use positions of Alphabet N° 5
3. International reference version of Alphabet N° 5
4. Graphical representation of control characters
5. Definitions and use of certain control characters
   - 5.1 Auxiliary alphabet (Code extension and expansion)

6. Implementation of the code on input/output media

7. Character sets for parallel data transmission

8. Others

Discussion and results

1. Possible modifications to Alphabet N° 5

1.1 Note 7 in position 2/3 and 2/4 of the code table

The ISO position was presented on the basis of resolution 4, doc. 97/2 n° 472 and of the guidelines for the preparatory group for the revision of ISO/R 646 (doc. 97/2 n° 471).
After much discussion and clarifications the following conclusions were reached by GM/ALP:

a) There is no need to indicate the additional symbols # and ¤ in parenthesis in position 2/3 and 2/4 respectively of the code table; the present method of indicating these symbols in the note seems to be clear enough. (Incidentally GM/ALP preferes to retain the name "number sign" for the symbol # and ¤ "currency sign" for the symbol as already used in CCITT Recommendation V.3 and also in ISO/R 1073-1969)

b) it is acceptable to combine note ② and ⑦ to the code table in a new note ② concerning both position 2/3 and 2/4

c) in the ISO proposal for the new note ② there is no provision for the necessary agreement between the parties concerned to choose a particular symbol among the symbols in question. Therefore a slightly modified text was prepared and is now submitted to ISO for their consideration.

② The symbol £ is assigned to position 2/3 and the symbol $ is assigned to position 2/4. In a situation where there is no requirement for the symbol £, the symbol # (number sign) may be used in position 2/3. Where there is no requirement for the symbol $, the symbol ¤ (currency sign) may be used in position 2/4. The chosen allocation of symbols to these positions for international information interchange should be agreed by the parties concerned.
It should be noted that, unless otherwise agreed between sender and recipient, the symbols £ and $ do not designate the currency of a specific country.

1.2 Additional preferred graphic characters for "national use" positions

The ISO proposal as stated in doc. 97/2 N° 396 was examined and favourably accepted. Consequently note 3 to the code table should be modified as follows:

③ Reserved for national use. These positions are primarily intended for alphabetic extensions.
If they are not required for that purpose, they may be used for symbols and a preferred choice is shown in parenthesis.

(NOTE: I noticed that there is a difference in the text between note ③ in Recommendation V.3 and in R 646. The phrase: "Some restrictions are placed on the use of these characters on the general telecommunication networks for international transmission" is not present in V.3. In the spirit of the revision agreed during the London meeting, I think that the preparatory working group should consider the deletion of this sentence.)

However, the Japanese Delegation stated some difficulties in accepting the symbol \ (reverse Solidus) to position 5/12 as a preferred symbol. According to the present Japanese Draft Standard, position 5/12 is used for the currency symbol ¥ (YEN), and the reverse solidus is allocated to position 6/0 replacing the grave accent. If the reverse solidus is assigned to position 5/12 as a preferred symbol, this can no longer be allocated to 6/0 in the Japanese standard due to the provision in the Explanatory Note 4.4. "Dual allocation" in R 646.

The Japanese delegation therefore requested that a decision of GM/ALP on this point be suspended until after the next meeting of ISO which may take up this point again.

(NOTE: the Japanese request being accepted, no final decision was taken on position 5/12. During the examination of the draft report I suggested that this item be further discussed during the Preparatory Working Group meeting in Geneva next January together with the Japanese delegation; if, as I think it will be possible, the problem will be finalized with the satisfaction of the Japanese national committee, there will be no further need to reopen the discussion at the GM/ALP level since the other delegations have unanimously agreed the ISO proposal).

2. National Use Positions

A CCITT document based on docs. 97/2 n° 358 and 382 was examined with interest and considered useful information. Some mistakes were reported by various delegates and the table correspondingly amended.

(NOTE: in respect of doc. 97/2 n° 474 recently issued by the secretariat, there are the following changes to be made:

Danemark	:	position 7/11	æ
		position 7/12	ø
Japan	:	position 6/0	\
Norway	:	position 5/11	Æ
		position 7/11	æ

Furthermore I was asked a question regarding the positions left blank on the table of national choices; are they left blank because they are not yet defined at the national level or are they to be considered according to the code table and the relevant notes in ISO/R 646 - 1967?
In the light of the revised R 646 and of the international version, my suggestion is that each member body could be requested to fill again table 1 in doc. 97/2 n° 474 clearly specifing the assignments to all the columns.)

3. International reference version

The ISO proposal for an "International Reference Version" of the 7-bit code was examined and discussed. Clarifications were made about its meaning and the general meaning of international interchange of data which seem to be interpreted in different ways by the various community interested in the subject: the industry, the users and the Administrations.
During the discussion certain administrations expressed their consent to such an international reference version.
Finally, the GM/ALP expressed the view that, at this initial stage of the study of this question in the ISO, the CCITT

cannot make a firm decision but would look favourably on the study of this question in the ISO within the framework of section 5, doc. 97/2 n° 471.

4. Graphical representation of Control characters:

Draft ISO Recommendation 2047 was examined and some differences were noticed among some graphical symbols represented there and those standardized for International Alphabet N° 2 in the CCITT (see CCITT recommendations S.4 and S.5).

Function	Alphabet N° 5	Alphabet N° 2
ENQ	⊠	✠
BEL	⌓	⍾
CR	∈	<

These graphical symbols used for Alphabet N° 2 have been standardized for many years and are well known internationally . They also should be used in the case of alphabet N° 5.

The GM/ALP, therefore, wishes that ISO would use at least the two symbols ✠ and ⍾ for positions 0/5 and 0/7 respectively.

(NOTE: The graphical representation of the CR function was accepted as in the ISO DR 2047 because of the possible confusion between the present symbol in alphabet N° 2 and the "less than" graphic (pos. 3/12) of the 7-bit code table.

5. Definitions and use of certain control characters:

This point of the agenda covers an item in the program of work of GM/ALP requiring further study on the definition and use of certain control characters, such as SO, SI, ESC, DLE etc.

Some contributions were examined in this connection:

a) Draft ISO Proposal for Code Independent Transmission Procedures (doc. ISO/TC97/SC6 n° 323).
This document defines the means by which data communication systems can transfer information without coding

restrictions. This is achieved by use of DLE sequences. The GM/ALP, in the framework of its terms of reference, examined this document as regards the compatibility with Recommendation V.3 and found no contradiction in this respect.

b) Part of a document submitted by the French administration indicating the need of new characters to control the new functions which are available from the visual display devices which could be used for telegraph operation in the future.
During the discussion it was pointed out that the Expert Group of ISO/TC97/SC2 dealing with code extension techniques is already considering the requirement of visual displays in connection with the additional control functions available through code extension.
The consensus of the meeting was in favour of this work and the GM/ALP concluded the discussion as follows:

- It has been agreed between ISO and CCITT that there is a need of additional functions for visual display machines.
- The study should be made in collaboration with ISO.
- These supplementary function codes might be obtained by means of the code extension procedures.

5.1 Auxiliary alphabet (Code extension and expansion procedures)

This item was dealt with in three successive steps.

5.1.1 Necessity of an auxiliary alphabet within CCITT

A question was raised by the Chairman as:

"Code combination 0/14 (Shift-Out) permits the introduction of auxiliary alphabets outside the standardized set table. Can a sufficiently widespread need be foreseen at present to warrant the standardization of a table of 95 auxiliary signs? If so, what should these signs be and what would be the resulting conditions for the apparatus using them?"

In reply to this question the GM/ALP was of the opinion that this subject is primarily concerned with the ISO and taking account of the present state of study on code extension and expansion technique in the ISO, there seems no need to make

standardization of a supplementary code in the CCITT.
However, the GM/ALP shall, of course, be interested in the principle of this study being carried out by the ISO.

5.1.2 Code extension and expansion procedures presently studied by the ISO

The following three ISO documents were available to the meeting in this connection:

- DR ISO 2022: Code extension procedures for the ISO 7-Bit code
- ISO/TC97/SC2 n° 399: Code Expansion and a structure for a family of 8-bit codes
- ISO/TC97/SC2 n° 414: Classes of three-characters Escape Sequences

The meeting was informed of the new development taken by ISO/TC97/SC2 at the last meeting in London. In absence of the new combined draft (currently 97/2 N° 473), GM/ALP examined the principles included in the three aforementioned documents and found no problem concerning the compatibility of the proposed methods with the implementation of Alphabet N° 5. GM/ALP will however wait with their final decision until the ISO proposal in its amended version will be available.

As to the definitions of three function characters "Shift-out", "Shift-in" and "Escape" which are used in the code extension technique, some discrepancy is seen on the present definition of the ESC character in Rec. V.3 (also ISO/R 646) and the actual mode of its application in the code extension procedures.

The GM/ALP wishes that this point should be studied by the ISO and if necessary, a modification of the existing definition of ESC character be submitted in the CCITT.

5.1.3 Registration of Escape sequences

The GM/ALP took note with interest of the document 97/2 n° 409 which specifies the method for the registration of escape sequences within the ISO and had no objection to this ISO internal procedure.

6. Implementation of the code on input/output media

It was stressed again at the meeting that it is not the role of CCITT to lay down the detail physical specifications of equipment.
GM/ALP considered then the existence of several ISO Recommendations on I/O media; particularly those regarding the punched tape are of interest of the Administrations.
No CCITT Recommandation will however be published, regarding the physical characteristics of the media. This is felt to be a national matter.

As for the implementation of the code on media an appropriate extract from ISO/R 1113 "Representation of 6 and 7 bit coded character sets on punched tape" will be published as an Annex to Recommendation V.3.

7. Character sets for parallel data transmission

The GM/ALP took note of a document from the ISO on this subject (97/2 n° 470).

The GM/ALP understood that this study in the ISO was in accordance with the point mentioned in CCITT Recommendation V.30, paragraph 10 character set. It is suggested in this paragraph that the elaboration of a recommendation for a character set to be used for parallel data transmission conforming to Rec. V.30 is primarily the task of the ISO in collaboration with the CCITT.

In the light of the fact that all the studies concerning parallel data transmission have been carried out by Special Study Group A in collaboration with the ISO, the GM/ALP is of the opinion that this question should also be taken up by Special A in the framework of Question 1/A - Point Q : Further study of parallel data transmission for universal use on telephone circuits.

8. Others

The GM/ALP took note of two documents from ECMA and ISO concerning keyboards.

These documents have been primarily published for study by Study Group VIII.

Conclusion:

In closing the meeting the chairman wished to express the thanks of the GM/ALP to ISO for the many technical contributions submitted to their consideration in the certainty that this fruitful cooperation will continue in the future.

Respectfully submitted

Mauro Ballabeni
ISO Representative
at the GM/ALP meeting

December 1970

X3L2/1075c

Association Française
de Normalisation
Tour EUROPE
CEDEX 7
92 - PARIS - LA DEFENSE
Tél. Paris (1) 788.11.11

I S O
INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
ORGANISATION INTERNATIONALE DE NORMALISATION

ISO/TC 97/SC 2
CHARACTER SETS AND CODING
Secretariat
AFNOR (France)

ISO/TC 97/SC 2
(Sec. 166)

November 1970

483E

REPORT JOINT WORKING PARTY
ON NEW ALPHABET

(Geneva - 18th-20th November 1970)

The "New Alphabet" Joint Working Party met in Geneva from 18 to 20 November 1970 under the chairmanship of Mr. G. BAGGENSTOS (Switzerland). The list of participants is given in the Annex to this Report.

During the meeting the GM/ALP examined a question which is common to Study Groups I, VIII and Special A, i.e. Question 6/1, a part of question 9/VIII and question 1/A - point C on the following points :

	Documents examined (GM/ALP - N°)
1. Possible modification to Alphabet n° 5	
1.1 Note 7 in positions 2/3 and 2/4 of the code table	1, 21
1.2 Additional preferred graphic characters for "national use" positions	1, 3, 21
2. National use positions of Alphabet n° 5	3, 4
3. International reference version of Alphabet n° 5	21
4. Graphical representation of control characters	2, 16

NMAH
311/9

	Documents examined (GM/ALP)
5. Definitions and use of certain control characters	8, 15 (para. II.5), 17 (para. 8)
5.1 Auxiliary alphabet (code extension and expansion)	6 (para. 8), 10 (+ 20), 11, 12, 13, 19 (para. 8), 23, (para. 8)
6. Implementation of the code on input/output media	6 (para. 5), 7, 15 (para. II.5)
7. Character sets for parallel data transmission	22
8. Others	5, 9, 14

1. Possible modification to Alphabet n° 5

1.1 Note 7 in positions 2/3 and 2/4 of the code table.

The GM/ALP considered carefully the I.S.O. proposal which appears in Doc. GM/ALP - n° 21, points 3.1 and 4 on this subject and reached the following conclusion :

a) Indication of the symbols # and ¤ in parenthesis in positions 2/3 and 2/4 respectively.

The GM/ALP is of the opinion that there is no need to indicate these additional symbols in parenthesis, because the present method of indicating these symbols in note 7 seems to be clear enough.

b) New note 2, superseding existing notes 2 and 7, proposed by the I.S.O.

The following text has been proposed by the I.S.O. :

"2) The symbol £ is assigned to position 2/3 and the symbol $ is assigned to position 2/4. In a situation where there is no requirement for the symbol £ the symbol # (number symbol) may be used in position 2/3. Where there is no requirement for the symbol $ the symbol ¤ (sun symbol) may be used in position 2/4. It should be noted that, unless otherwise agreed between sender and recipient, the symbols £, $ or ¤ do not designate the currency of a specific country."

i) the GM/ALP considers it acceptable to combine the existing notes 2 and 7 concerning positions 2/3 and 2/4 of Alphabet n° 5, provided that no substantial change to these existing notes will be made.

ii) The GM/ALP then examined whether the version of note 2 proposed by the I.S.O. is acceptable to the C.C.I.T.T. During the discussion the following points were raised by certain delegates :

- In the I.S.O. proposal there is no provision for the necessary agreement between the parties concerned to choose a particular symbol among the symbols in question ;
- in the present note 7 to Alphabet n° 5, priority is given to the symbols £ in position 2/3 and $ in position 2/4 for the international information interchange. However, in the view of the I.S.O. these four symbols should be placed on an equal footing without giving any priority to any symbol. This would lead to a substantial change of Recommendation V.3.

Taking into account the above-mentioned points as well as the I.S.O. proposal, the GM/ALP has prepared the following new note 2 which could supersede existing notes 2 and 7 with a view to forwarding it to the I.S.O. for their consideration :

"The symbol £ is assigned to position 2/3 and the symbol $ is assigned to position 2/4. In a situation where there is no requirement for the symbol £ the symbol # (number symbol) may be used in position 2/3. Where there is no requirement for the symbol $ the symbol ¤ (currency sign) may be used in position 2/4. The chosen allocations of symbols to these positions for international information interchange should be agreed by the parties concerned. It should be noted that, unless otherwise agreed between sender and recipient, the symbols £, $ or ¤ do not designate the currency of a specific country."

It is recommended that the C.C.I.T.T. Secretariat convey the above wording to the I.S.O. for their consideration in the re-drafting of their Recommendation R.646.

With this modification to notes 2 and 7, existing note 8 would be renumbered as note 7. The code table of Alphabet n° 5 should also be modified accordingly.

1.2 Additional preferred graphic characters for "national use" positions

The I.S.O., in Document GM/ALP-N° 21, proposes to incorporate preferred graphics to positions 5/12 (\), 7/11 ({), 7/12 (|) and 7/13 (}) such as it is already done for positions 4/0, 5/11 and 5/13 which show assigned preferred graphics.

A consensus of opinion in the GM/ALP was in favour of this I.S.O. proposal and the present note 3 to the code table should accordingly be modified as follows :

> "3) Reserved for national use. These positions are primarily intended for alphabetic extensions. If they are not required for that purpose, they may be used for symbols and a preferred choice is shown in parenthesis."

...

However, the Japanese delegation stated some difficulty in accepting the symbol\ (reverse solidus) to position 5/12 as a preferred symbol. According to the present Japanese national version, position 5/12 is used for their national currency sign and position 6/0 for the reverse solidus. If the reverse solidus is assigned to position 5/12 as a preferred symbol, position 6/0 can no longer be used for the reverse solidus according to the provision in 1.8 dual allocation of Recommendation V.3 (also I.S.O. /R.646).

The Japanese delegation therefore requested that a decision of the GM/ALP on this point be suspended until after the next meeting of the I.S.O. which may take up this point again.

The GM/ALP acceded the Japanese request in respect of position 5/12 but accepted the I.S.O. proposal for positions 7/11, 7/12 and 7/13.

2. National use position

The GM/ALP noted Documents GM/ALP n° 3 and 4 presented by the I.S.O. and the E.C.M.A. respectively.

The allotment of characters to these national positions is a national matter and they have been published for information only.

The GM/ALP looked at these documents with interest and also with gratitude to these two organizations. The I.S.O. representative mentioned that the information given in Document GM/AL n° 3 was still far from being satisfactory and the I.S.O. is in the process of collecting further information from their national member bodies. The C.C.I.T.T. will be kept informed by the I.S.O. on this subject.

Meanwhile, some corrections to Tables 1 and 2 of Document GM/ALP n° 3 have been made and these corrected tables will be published as a C.C.I.T.T. contribution.

3. International reference version

The GM/ALP examined the I.S.O. proposal on an international reference version which was mentioned in Document GM/ALP n° 21. This version is intended for general use in all cases when there is no requirement to use a national version or any other "application oriented" character set.

During the discussion certain administrations expressed their consent to such an international reference version.

The GM/ALP expressed the view that, at this initial stage of the study of this question in the I.S.O., the C.C.I.T.T. cannot make a firm decision but would look favourably on the study of this question in the I.S.O. within the framework of the Annex to Document GM/ALP n° 21.

....

ANNEXE - ANNEX

LISTE DES PARTICIPANTS

LIST OF PARTICIPANTS

Président :
Chairman : G/ BAGGENSTOS (Suisse / Switzerland)

AUSTRALIE/AUSTRALIA	R.A.R. BROWN R. KERR N.R. CRANE
CUBA	J.L. CARRENO J. HERNANDEZ H.
FRANCE	A. DUPONT C. MOUTON
ITALIE/ITALY	U. DEGANO
JAPON/JAPAN	T. OTA
NORVEGE/NORWAY	T.E. MOE
PAYS-BAS/NETHERLANDS	J.L. VAN DALE
PORTUGAL	M.J.M. ATHAYDE J.C.B. CORREIA
R.F. d'ALLEMAGNE F.R. of GERMANY	H. GABLER F. BARDUA
SUISSE/SWITZERLAND	F. KELLER
SUEDE/SWEDEN	E. OSTERLUND
U.R.S.S./U.S.S.R.	N. ETROUKINE A. OGOCHKINE E. BASILEVITSCH V. ILINE Mme M. TOKAREVA Mme A. VOLKOVA

Exploitations privées reconnues
Private Operating Agencies

A.T. & T.	E.W. KETTNICH
Italcable	R. DE LONGIS
Kokusai Denshin Denwa	S. SHIMURA Y. TAKAHASHI
Nippon Telegraph and Telephone Public Corp.	K. MURAMATSU T. MATSUMOTO
Nigerian External Telecommunications Ltd.	S.O. OGUNLEYE O.A. COKER
U.K. Post Office	G.W. ADAMS F.W. DAVIES

Organismes scientifiques ou industriels
Scientific or Industrial Organizations

I.T.T.	J.V. EVANS
S.A.G.E.M.	P.N. LEGER
Standard Elektrik Lorenz A.G.	B. CRAMER

Organisations internationales
International Organizations

I.P.T.C.	F.I. RAY
I.S.O.	M.F. BALLABENI
Secrétariat du C.C.I.R.	A. ARAI
Secrétariat du C.C.I.T.T.	E. HUMMEL, Conseiller supérieur T. OKABE, Ingénieur Mlle M. LANG STEVENSON, Assistante

X3L2/1055
1970 Nov. 24

AMERICAN NATIONAL STANDARDS INSTITUTE, INC. - COMMITTEE CORRESPONDENCE

Reply to: Eric H. Clamons
RD 1
North Wales, Pa. 19454
(215)643-1300, X2552/3

1970 November 24

TO: X3L2

SUBJECT: LETTER BALLOT ON MODIFICATIONS TO R646

RE: Document X3L2/1041

As announced at the last meeting of X3L2, this letter ballot is sent to you in preparation for the Preparatory Group Meeting in January 1971 to which Dr. Blanton C. Duncan was elected our representative. The questions have been addressed specifically to modification of R646 and allow you to express your wish regarding changes in X3.4-1968. If you have additional comments, please attach them.

As a technical member of X3L2, you are required to answer all questions to the best of your ability; you may not abstain. Only one ballot per voting unit is permitted, i.e. the member or his alternate, not both, may vote.

Eric H Clamons

Eric H. Clamons
Chairman X3L2

EHC/jas

Att.

NMAH 311/2

YES	NO	YES	NO

I. GENERAL DIFFERENCES

A. The so-called "International Reference Version of the ECMA 7 Bit Coded Character Set" is included as a version for general use during international interchange situations. X3.4-1968 does not include this as part of the ANSI Standard. There are specific assignments in that table (Table 2) which do not agree with ASCII which are:

Col/Row	International Version	ASCII		YES	NO	YES	NO
2/4	¤	$		○	○	○	○
[redacted]							
7/12	|	¦	(minor diff.)	○	○	○	○

It also does not agree with the ECMA "Standard" version in the following positions:

Col/Row	International	ECMA-6	
2/3	#	£	
2/4	¤	$	
[redacted]			
7/14	~	—	(minor diff.)

B. In general, the meanings of most all controls (columns 0 and 1) are specifically spelled out. The first sentence of each definition normally agrees with each ASCII definition, but then ECMA-6 continues to further specify each control's function in detail. (Specific differences are given in Section III.)

C. Notes on specific substitutions for graphics, application of the code(s), and structure of controls and graphics are all included as parts of the ECMA-6 standard. X3.4-1968, however, specifically states that such material is not included as part of the standard, but is only suggested as aids to the user in appendix material of the Standard.

II. GRAPHICS

A. Col/Row 2/3 main choice is '£' in ECMA-6 with an alternate being '#'. X3.4-1968 main choice and alternate choice are reversed. ○ ○ ○ ○

B. Col/Row 2/4 allows '¤' as a specific alternate choice. A3.4-1968 allows no specific alternate choice. ○ ○ ○ ○

		R646		X3.4	
		YES	NO	YES	NO
C.	The characters '10' and '11' are allowed as optional use characters to replace ' : ' and ' ; ' respectively in ECMA-6. Again, no specific optional choices for these two characters are allowed in ASCII.	○	○	○	○
D.	ASCII overline is '~', which also means tilde (diacritical). ECMA-6 uses ' — '. However, in both standards, stylization options are duly noted. (Similarly for positions 5/14 and 6/0.)	○	○	○	○
E.	The method of utilizing 4 positions which are sometimes stand-alone graphics and other times diacritical marks (2/2 ", 2/7 ', 2/12 , and 5/14 ∧) is somewhat different in ECMA-6 in that the exact use of BS (backspace) is defined as the 3 character sequence: character - BS - diacritical. No such use is defined or inferred by X3.4-1968.	○	○	○	○
III.	CONTROLS				
A.	Transmission Controls are called Communication Controls in X3.4-1968 (a minor difference).	○	○	○	○
B.	X3.4-1968 specifically designates that the 4 Information Separators (FS, GS, RS and US) are to be used only in hierarchical order. ECMA-6 also allows for other uses (but does not specifically say what those uses would be).	○	○	○	○
C.	Format Effectors are defined by specifically spelling out the concepts of "printed page", "character positions on a line", and the "active position" during processing. X3.4-1968 does not do this. (See also, the use of BS, Section II, E of this paper.)	○	○	○	○
D.	Device Controls (DC1, DC2, DC3, and DC4) are all given specific preferred meanings in ECMA-6. Only DC4 is spelled out in X3.4-1968, which agrees with the definition in ECMA-6.	○	○	○	○
E.	BEL - X3.4-1968 specifically calls for human intervention, ECMA-6 does not.	○	○	○	○
F.	DEL - ECMA-6 has an additional comment above and beyond X3.4-1968 definition in that DEL may also serve as media- or time-fill character, thus affecting the information layout and/or control of equipment. X3.4-1968 allows only NUL for this purpose.	○	○	○	○
G.	CAN - The Cancel doctrine is spelled out in ECMA-6, and is not in X3.4-1968. However, since this is the subject of a separate report it is not discussed here.				

		R646		X3.4	
		YES	NO	YES	NO
H.	DLE - ECMA-6 specifically requires that only graphics and/or other transmission controls may be part of a transmission escape sequence. X3.4-1968 does not specify any.	○	○	○	○
I.	ENQ - The use of ENQ to mean "WRU" as first use and possible later uses is not spelled out in X3.4-1968.	○	○	○	○
J.	ESC - The wording of the X3.4-1968 definition is much more general than that in ECMA-6 (specifically the second ECMA-6 paragraph on page 13). Since this control is the subject of an X3L2 experts group, which is working toward a standard for code extension, no more discussion in this paper is necessary.	○	○	○	○
K.	ETX - The X3.4-1968 definition is more detailed than ECMA-6 in that its definition is tied in with that of STX.	○	○	○	○
L.	LF - The fact that LF may also be used as a control which combines CR with itself (NL) is not specified on page 14, although it is defined elsewhere.	○	○	○	○
M.	NUL - As in DEL, ECMA-6 here again mentions that this character may be inserted into or removed from a data stream, thereby possibly affecting layout and control. This is not mentioned in X3.4-1968.	○	○	○	○
N.	SO - ECMA-6 specifies that in the Shift out state, controls (columns 0 and 1) and DEL will retain their meanings. This is not so stated in X3.4-1968.	○	○	○	○
O.	SOH - "Heading" is not defined by ECMA-6, although it is in X3.4-1968.	○	○	○	○
P.	SUB - The concept of this character is ill-defined in both standards. ECMA-6 adds the statement, "SUB is introduced by the processing or transmission system."	○	○	○	○
Q.	STX - The definition of this character is not (in ECMA-6) as specifically defined as it is in X3.4-1968 in that: (1) the terms "text" and "heading" are un-defined, and (2) the definition does not tie in its use with either SOH or ETX.	○	○	○	○
R.	SYN - The ECMA-6 definition limits the use of this control to "terminal equipments." X3.4-1968 does not.	○	○	○	○

UDC 681.3.042 : 003.62

Ref. No.: ISO / R 646 - 1971

X3L2/1072

ISO

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION

Delegates Notes

ISO RECOMMENDATION

R 646

7 BIT CODED CHARACTER SETS FOR INFORMATION PROCESSING INTERCHANGE

2nd EDITION

...... 1971

COPYRIGHT RESERVED

The copyright of ISO Recommendations and ISO Standards belongs to ISO Member Bodies. Reproduction of these documents, in any country, may be authorized therefore only by the national standards organization of that country, being a member of ISO.

For each individual country the only valid standard is the national standard of that country.

NMAH 311/4

Printed in Switzerland

Also issued in French and Russian. Copies to be obtained through the national standards organizations.

7 BIT CODED CHARACTER SET
FOR INFORMATION PROCESSING INTERCHANGE

1. GENERAL

1.1 Scope

1.1.1 This ISO Recommendation contains a set of 128 characters (control characters and graphic characters such as letters, numerals and symbols) with their coded representation. Most of these characters are mandatory and unchangeable, but provision is made for some flexibility to accomodate special national and other requirements.

1.1.2 The need for graphics and controls in data processing and in data transmission has been taken into account in determining this character set.

1.1.3 This ISO Recommendation consists of a general table with a number of options, notes, a legend and explanatory notes. It also contains a specific International Reference Version, guidance on the exercise of the options to define specific national versions and application oriented versions.

1.1.4 This character set is primarily intended for the interchange of information among data processing systems and associated equipment.

1.1.5 This character set is applicable to all Latin alphabets.

1.1.6 The character set includes facilities for extension where its 128 characters are insufficient for particular applications.

1.2 Implementation

1.2.1 This character set should be regarded as a basic alphabet in abstract sense. Its practical use requires definitions of its implementation in various media. For example, this could include punched tapes, punched cards, magnetic tapes and transmission channels, thus permitting interchange of data to take place either indirectly by means of an intermediate recording in a physical medium, or by local electrical connection of various units (such as input and output devices and computers) or by means of data transmission equipment.

1.2.2 The implementation of this coded character set in physical media and for transmission, taking into account the need for error checking, is the subject of other ISO Re-

---

2. TABLE I

Bits				b_7	0	0	0	0	1	1	1	1
				b_6	0	0	1	1	0	0	1	1
				b_5	0	1	0	1	0	1	0	1
b_4	b_3	b_2	b_1	Column Row	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	TC$_7$(DLE)	SP	0	③	P	` ④	p
0	0	0	1	1	TC$_1$ (SOH)	DC$_1$	!	1	A	Q	a	q
0	0	1	0	2	TC$_2$ (STX)	DC$_2$	" ⑥	2	B	R	b	r
0	0	1	1	3	TC$_3$ (ETX)	DC$_3$	② £ (#)	3	C	S	c	s
0	1	0	0	4	TC$_4$ (EOT)	DC$_4$	② $ (¤)	4	D	T	d	t
0	1	0	1	5	TC$_5$ (ENQ)	TC$_8$(NAK)	%	5	E	U	e	u
0	1	1	0	6	TC$_6$ (ACK)	TC$_9$(SYN)	&	6	F	V	f	v
0	1	1	1	7	BEL	TC$_{10}$(ETB)	´ ⑥	7	G	W	g	w
1	0	0	0	8	FE$_0$ (BS)	CAN	(	8	H	X	h	x
1	0	0	1	9	FE$_1$ (HT)	EM	)	9	I	Y	i	y
1	0	1	0	10	FE$_2$ (LF)①	SUB	*	: ⑦	J	Z	j	z
1	0	1	1	11	FE$_3$ (VT)①	ESC	+	; ⑦	K	③	k	③
1	1	0	0	12	FE$_4$ (FF)①	IS$_4$ (FS)	, ⑥	<	L	③	l	③
1	1	0	1	13	FE$_5$ (CR)①	IS$_3$ (GS)	-	=	M	③	m	③
1	1	1	0	14	SO	IS$_2$ (RS)	.	>	N	^ ④⑥	n	— ④⑤
1	1	1	1	15	SI	IS$_1$ (US)	/	?	O	_	o	DEL

NOTES ABOUT THE TABLE

① The Format Effectors are intended for equipment in which horizontal and vertical movements are effected separately. If equipment requires the action of CARRIAGE RETURN to be combined with a vertical movement, the Format Effector for that vertical movement may be used to effect the combined movements. For example, if NEW LINE (CR + LF) is required, FE_2 will be used to represent it. This substitution requires agreement between the sender and the recipient of the data.
The use of these combined functions is not allowed for international transmission on general switched telecommunication networks (telegraph and telephone networks).

② The symbol £ is assigned to position 2/3 and the symbol $ is assigned to position 2/4. In a situation where there is no requirement for the symbol £ the symbol # (number sign) may be used in position 2/3. Where there is no requirement for the symbol $ the symbol ¤ (currency sign) may be used in position 2/4. The chosen allocations of symbols to these positions for international information interchange should be agreed by the parties concerned. It should be noted that, unless otherwise agreed between sender and recipient, the symbols £, $ or ¤ do not designate the currency of a specific country.

③ National Use Positions. The allocations of characters to these positions lies within the responsibility of National Standardization Bodies. These positions are primarily intended for alphabet extensions. If they are not required for that purpose, they may be used for symbols.

④ Position 5/14, 6/0 and 7/14 are provided for the diacritical signs CIRCUMFLEX ACCENT, GRAVE ACCENT and OVERLINE. However, these positions may be used for other graphical characters when it is necessary to have 8, 9 or 10 positions for national use.

⑤ Position 7/14 is used for the graphic character ‾ (OVERLINE), the graphical representation of which may vary according to national use to represent ˜ (TILDE) or another diacritical sign provided that there is no risk of confusion with another graphic character included in the table.

⑥ The graphic characters in positions 2/2, 2/7, 2/12, 5/14 have respectively the significance of QUOTATION MARK, APOSTROPHE, COMMA and UPWARD ARROW HEAD; however, these characters take on the significance of DIARESIS, ACUTE ACCENT, CEDILLA and CIRCUMFLEX ACCENT diacritical signs when they are preceded or followed by BACKSPACE.

3. LEGEND

3.1 CONTROL CHARACTERS

Abbreviation	Note	Meaning	Position in the Code Table
ACK		Acknowledge	0/6
BEL		Bell	0/7
BS		Backspace	0/8
CAN		Cancel	1/8
CR	1	Carriage Return	0/13
DC		Device Control	-
DEL		Delete	7/15
DLE		Data Link Escape	1/0
EM		End of Medium	1/9
ENQ		Enquiry	0/5
EOT		End of Transmission	0/4
ESC		Escape	1/11
ETB		End of Transmission Block	1/7
ETX		End of Text	0/4
FE		Format Effector	-
FF	1	Form Feed	0/12
FS		File Separator	1/12
GS		Group Separator	1/13
HT		Horizontal Tabulation	0/9
IS		Information Separator	-
LF	1	Line Feed	0/10
NAK		Negative Acknowledge	1/5
NUL		Null	0/0
RS		Record Separator	1/14
SI		Shift-In	0/15
SO		Shift-Out	0/14
SOH		Start of Heading	0/1
SP		Space (see Section 6.2)	2/0
SUB		Substitute Character	1/10
STX		Start of Text	0/2
SYN		Synchronous Idle	1/6
TC		Transmission Control	-
US		Unit Separator	1/15
VT	1	Vertical Tabulation	0/11

3.2 GRAPHIC CHARACTERS

Graphic	Note	Name	Position in the Code Table
(space)		A normally non-printing graphic character (see Section 6.2)	2/0
!		Exclamation mark	2/1
"	6	Quotation mark, Diaresis	2/2
£	2	Pound sign	2/3
#	2	Number sign	2/3
$	2	Dollar sign	2/4
¤	2	Currency sign	2/4
%		Percent	2/5
&		Ampersand	2/6
'	6	Apostrophe, Acute Accent	2/7
(		Left Parenthesis	2/8
)		Right Parenthesis	2/9
*		Asterisk	2/10
+		Plus sign	2/11
,	6	Comma, Cedilla	2/12
-		Hyphen, Minus sign	2/13
.		Full stop (period)	2/14
/		Solidus	2/15
:	7	Colon	3/10
;	7	Semi-colon	3/11
<		Less than	3/12
=		Equals	3/13
>		Greater than	3/14
?		Question mark	3/15
^	4 6	Upward arrow head, Circumflex accent	5/14
_		Underline	5/15
`	4	Grave accent	6/0
‾	4 5	Overline, Tilde	7/14

4. EXPLANATORY NOTES

4.1 Numbering of the positions in Table I

Within any one character the bits are identified by b_7, b_6 ... b_1, where b_7 is the highest order, or most significant bit, and b_1 is the lowest order, or least significant bit.

If desired these may be given a numerical significance in the binary system, thus:

Bit identification:	b_7	b_6	b_5	b_4	b_3	b_2	b_1
Significance:	64	32	16	8	4	2	1

In the code table the columns and rows are identified by numbers written in binary and decimal notations.

Any one position in the code table may be identified either by its bit pattern, or by its column and row numbers. For instance, the position containing the figure 1 may be identified:

- by its bit pattern, e.g. 011 0001
- by its column and row numbers, e.g. 3/1.

4.2 Diacritical signs

In the character set, some printing symbols may be designed to permit their use for the composition of accented letters when necessary for general interchange of information. A sequence of three characters, comprising a letter, BACKSPACE and one of these symbols, is needed for this composition, and the symbol is then regarded as a diacritical sign. It should be noted that these symbols take on their diacritical significance when they are preceded or followed by one BACKSPACE character; for example, the symbol corresponding to the code combination 2/7 (') normally has the significance of APOSTROPHE, but becomes the diacritical sign ACUTE ACCENT when it precedes or follows a BACKSPACE character.

In order to increase efficiency, it is possible to introduce accented letters (as single characters) in the positions marked by note ③ in the code table. According to national requirements, these positions may contain special diacritical signs.

4.3 Names, Meanings and Fonts of Graphic Characters

This Recommendation assigns at least one name to denote each of the graphic characters displayed in Tables I and II. The names chosen to denote graphic characters, are intended to reflect their customary meanings, however, this Recommendation does not define and does not restrict the meanings of graphic characters. In addition, this Recommendation does not specify a particular style or font design for graphic characters.

Under the provision of Note 3 of the Table I graphic characters which are different from the characters of the International Reference Version may assigned to the National Use Positions. When such assignments are made the graphic characters should have distinct forms and be given distinctive names which are not in conflict with any of the forms or the names of any of the graphic characters in the International Reference Version.

4.4 Uniqueness of Character Allocation

A character allocated to a position in the Table I may not be placed elsewhere in the table. For example, in the case of position 2/3 the character not used cannot be placed elsewhere. In particular the POUND sign (£) can never be represented by the bit combination of position 2/4.

5. VERSIONS OF TABLE I

5.1 General Remarks

5.1.1 In order to use Table I for information interchange, it is necessary to exercise the options left open. A single character must be allocated to each of the positions for which this freedom exists. A code table completed in this way is called a VERSION.

5.1.2 The contents of the Notes to Table I, of the Explanatory Notes and of the Legend apply fully to any version.

5.2 International Reference Version

In international information processing interchange the International Reference Version (Table II) is postulated unless a particular agreement exists between sender and recipient of the data.

Legend

@	Commercial at	4/0
[	Left square bracket	5/11
\	Reverse solidus	5/12
]	Right square bracket	5/13
{	Left curly bracket	7/11
|	Vertical line	7/12
}	Right curly bracket	7/13

It should be noted that no substitution is allowed when

TABLE II

Bits				b7	0	0	0	0	1	1	1	1
				b6	0	0	1	1	0	0	1	1
				b5	0	1	0	1	0	1	0	1
b4	b3	b2	b1	Row \ Column	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	TC$_7$(DLE)	SP	0	@	P	`	p
0	0	0	1	1	TC$_1$ (SOH)	DC$_1$	!	1	A	Q	a	q
0	0	1	0	2	TC$_2$ (STX)	DC$_2$	"	2	B	R	b	r
0	0	1	1	3	TC$_3$ (ETX)	DC$_3$	#	3	C	S	c	s
0	1	0	0	4	TC$_4$ (EOT)	DC$_4$	¤	4	D	T	d	t
0	1	0	1	5	TC$_5$ (ENQ)	TC$_8$(NAK)	%	5	E	U	e	u
0	1	1	0	6	TC$_6$ (ACK)	TC$_9$(SYN)	&	6	F	V	f	v
0	1	1	1	7	BEL	TC$_{10}$(ETB)	'	7	G	W	g	w
1	0	0	0	8	FE$_0$ (BS)	CAN	(	8	H	X	h	x
1	0	0	1	9	FE$_1$ (HT)	EM	)	9	I	Y	i	y
1	0	1	0	10	FE$_2$ (LF)	SUB	*	:	J	Z	j	z
1	0	1	1	11	FE$_3$ (VT)	ESC	+	;	K	[	k	{
1	1	0	0	12	FE$_4$ (FF)	IS$_4$ (FS)	,	<	L	\	l	|
1	1	0	1	13	FE$_5$ (CR)	IS$_3$ (GS)	-	=	M	]	m	}
1	1	1	0	14	SO	IS$_2$ (RS)	.	>	N	^	n	—
1	1	1	1	15	SI	IS$_1$ (US)	/	?	O	_	o	DEL

5.3 National Versions

5.3.1 The responsibility for defining National Versions lies within the National Standardization Bodies. These Bodies have to exercise the options available and to make the required selection.

5.3.2 If so required, more than one national version can be defined within a country. The different versions must then be well identified. In particular when for a given National Use Position, e.g. 5/12 or 6/0, alternative characters are required, this means that two different versions must be identified, even if they differ only by this single character.

5.3.3 If there is in a country no special demand for specific characters, it is strongly recommended that the characters of the International Reference Version be allocated to the same National Use Positions.

Note: Appendix XYZ shows, for information only, the allocation of characters to National Use Positions in different countries, so far they are known as per ...., 1971. However, it should be noted that the allocations appearing in the specifications of the official standards are the only valid ones.

5.4 Application Oriented Versions

5.4.1 Within national or international industries, organizations or professional groups application oriented versions can be used. They require precise agreement among the parties concerned, who will have to exercise the options available and to make the required selection.

6. FUNCTIONAL CHARACTERISTICS RELATED TO CONTROL CHARACTERS

Some definitions in this section are stated in general terms and more explicit definitions of use may be needed for specific implementation of the code table on recording media or on transmission channels. These more explicit definitions and the use of these characters are the subject of other ISO Recommendations.

6.1 General designations of Control Characters

The general designation of Control Characters involves a specific name followed by a subscript number.

They are defined as follows:

TC. Transmission Controls

Functional characters intended to control or facilitate transmission of information over telecommunication networks.

The use of the TC characters on the general telecommunication networks is the subject of other ISO Recommendations.

FE Format Effectors

Functional characters mainly intended for the control of the layout and positioning of information on printing and/or display devices. In the definitions of specific Format Effectors any reference to printing devices should be interpreted as including display devices.

The definitions of Format Effectors use the following concept.

i) The printed page is composed of a number of lines of characters.

ii) The characters forming a line occupy a number of positions called: character positions.

iii) The active position is that character position in which the character about to be processed would appear, if it were to be printed. The active position normally advances one character position at a time.

DC Device Control

Functional characters for the control of a local or remote ancillary device (or devices) connected to a data processing and/or telecommunication system. The controls are not intended to control the telecommunication systems; the latter should be achieved by the use of TCs.

Certain preferred uses of the individual DCs are given in Section 6.2.

IS Information Separators

Functional characters which are used to separate and qualify data logically. There are four such characters. They may be used either in hierarchical order or non-hierarchically; in the latter case their specific meaning depends on their applications.

If they are used hierarchically the ascending order is:

US, RS, GS, IS.

In this case data normally delimited by a particular separator cannot be split by a higher order separator but will be considered as delimited by any higher order separator.

6.2 Specific Controls and other non graphic characters

Individual members of the classes of controls are sometimes referred to by abbreviated class name and a subscript number (e.g. TC5) and sometimes by a specific name indicative of their use (e.g. ENQ). These controls, those which do not fall into classes, and other non graphic characters are defined below.

Different but related meaning may be associated with some of the controls but this would normally require agreement between sender and recipient of the data.

ACK ACKNOWLEDGE

A Transmission Control character transmitted by a receiver as an affirmative response to the sender.

BEL BELL

A character for use when there is a need to call for attention; it may control alarm or attention devices.

BS BACKSPACE

A Format Effector which moves the active position one character position backwards on the same line.

CAN CANCEL

The first character of a two character sequence indicating that the data associated with it is in error or is to be ignored.

This data includes the CAN sequence itself and all data previous to it up to, but excluding, the nearest previous occurrence of the character following CAN. A CAN sequence is always to be interpreted as if any previously occurring CAN sequence had already been completely processed.

In certain circumstances some control character may not be permitted to follow CAN or there may be a limit to the amount or type of data able to be cancelled.

Examples :

```
                    C
1)  A B X C D E X G A E F G         yields  A B X C D E F G
                    N

                    C   C
2)  A B X C D E X B A E A B C D     yields  A B C D
                    N   N

          C
3)  A B C A C D E                   yields  A B C D E
          N

          E       C                               E
4)  A B C S B D E A B               yields  A B C S B
          C       N                               C
```

CR CARRIAGE RETURN

A Format Effector which moves the active position to the first character position on the same line.

DEVICE CONTROLS

DC1 A device Control Character which is primarily intended for turning on or starting an ancillary device. If it is not required for this purpose, it may be used to restore a device to the basic mode of operation (see also DC2 and DC3), or for any other device control function not provided by other DCs.

DC2 A Device Control Character which is primarily intended for turning on or starting an ancillary device. If it is not required for this purpose, it may be used to set a device to a special mode of operation (in which case DC1 is used to restore the device to the basic mode), or for any other device control function not provided by other DCs.

DC3 A Device Control Character which is primarily intended for turning off or stopping an ancillary device. This function can be a secondary level stop, e.g. wait, pause, stand-by or halt (in which case DC1 is used to restore normal operation). If it is not required for this purpose, it may be used for any other device control function not provided by other DCs.

DC4 A Device Control Character which is primarily intended for turning off, stopping or interrupting an ancillary device. If it is not required for this purpose, it may be used for any other device control function not provided by other DCs.

Examples of use of the Device Controls:

1) One switching		
	on - DC2	off - DC4
2) Two independent switchings		
First one	on - DC2	off - DC4
Second one	on - DC1	off - DC3
3) Two dependent switchings		
general	on - DC2	off - DC4
particular	on - DC1	off - DC3
4) Input and Output switching		
Output	on - DC2	off - DC4
Input	on - DC1	off - DC~~4~~3

DEL DELETE

A character used primarily to erase or obliterate erroneous or unwanted characters in punched tape. DEL characters may also serve to accomplish media-fill or time-fill. They may be inserted into or removed from a stream of data without affecting the information content of that stream but then the addition or removal of these characters may affect the information layout and/or the control of equipment.

DLE DATA LINK ESCAPE

A transmission control character which will change the meaning of a limited number of contiguously following characters. It is used exclusively to provide supplementary data transmission control functions. Only graphics and Transmission Control Characters can be used in DLE sequences.

EM END OF MEDIUM

A control character which may be used to identify the physical end of the medium, or the end of the used, or wanted, portion of information recorded on a medium. The position of this character does not necessarily correspond to the physical end of the medium.

ENQ ENQUIRY

A Transmission Control Character used as a request for a response from a remote station - the response may include station identification and/or station status. When a "Who are you" function is required on the general switched transmission network, the first use of ENQ after the connection is established shall have the meaning "Who are you" (station identification). Subsequent use of ENQ may, or may not, include the function "Who are you", as determined by agreement.

EOT END OF TRANSMISSION

A Transmission Control Character used to indicate the conclusion of the transmission of one or more texts.

ESC ESCAPE

A functional character which is used to provide additional control functions. It alters the meaning of a limited number of contiguously following bit combinations. The use of this character is specified in ISO Recommendation R ...

ETB END OF TRANSMISSION BLOCK

A Transmission Control Character used to indicate the end of a transmission block of data where data is divided into such blocks for transmission purposes.

ETX END OF TEXT

A Transmission Control Character which terminates a text.

FF FORM FEED

A Format Effector which advances the active position to

HT HORIZONTAL TABULATION

A Format Effector which advances the active position to the next predetermined character position on the same line.

INFORMATION SEPARATORS

IS_1 (US) A character used to separate and qualify data logically; its specific meaning has to be defined for each application. If this character is used in hierarchical order as specified in the general definition of IS, it delimits a data item called a UNIT.

IS_2 (RS) A character used to separate and qualify data logically; its specific meaning has to be defined for each application. If this character is used in hierarchical order as specified in the general definition if IS, it delimits a data item called a RECORD.

IS_3 (GS) A character used to separate and qualify data logically; its specific meaning has to be defined for each application. If this character is used in hierarchical order as specified in the general definition of IS, it delimits a data item called a GROUP.

IS_4 (FS) A character used to separate and qualify data logically; its specific meaning has to be defined for each application. If this character is used in hierarchical order as specified in the general definition of IS, it delimits a data item called a FILE.

LF LINE FEED

A Format Effector which advances the active position to the same character position on the next line.

NAK NEGATIVE ACKNOWLEDGE

A Transmission Control Character transmitted by a receiver as a negative response to the sender.

NUL NULL

A character used to accomplish media-fill or time-fill. NUL characters may be inserted into or removed from a stream of data without affecting the information content of that stream but then the addition or removal of these characters may affect the information layout and/or the control of equipment.

SI SHIFT-IN

A functional character which is used in conjunction with SHIFT-OUT to extend the graphic character set of the code.

It reinstates the standard meanings of the bit combinations which follow it. The effect of this character is described in R ...

SO SHIFT-OUT

A functional character which is used in conjunction with SHIFT-IN to extend the graphic character set of the code. It alters the meaning of the bit combinations of columns 2 to 7 which follow it until a SHIFT-IN character is reached. However, the characters SPACE (2/0) and DELETE (7/15) are unaffected by SHIFT-OUT. The effect of this character when using Code Extension Techniques is described in R ...

SOH START OF HEADING

A Transmission Control Character used as the first character of a heading of an information message.

SP SPACE

A layout character which advances the active position one character position on the same line.
This character can also be regarded as a non-printing graphic.

STX START OF TEXT

A Transmission Control Character which precedes a text and which is used to terminate a heading.

SUB SUBSTITUTE CHARACTER

A character used to replace a character which has been found to be invalid or in error. SUB is introduced by the processing system.

SYN SYNCHRONOUS IDLE

A Transmission Control Character used by a synchronous transmission system in the absence of any other character (idle condition) to provide a signal from which synchronism may be achieved or retained between data terminal equipments.

VT VERTICAL TABULATION

A Format Effector which advances the active position to the same character position on the next pre-determined line.

TELETYPE CORPORATION

5555 TOUHY AVENUE SKOKIE, ILLINOIS 60076
PHONE 982-2000 AREA CODE 312 TWX 910-223-3611

April 28, 1971

MR. JOHN G. FLETCHER
Lawrence Radiation Laboratory
P.O. Box 808
Livermore, California 94550

Dear John:

I must apologize for not being able to participate with you on 3rd and 4th of May, but as you know, I had to leave for my European meetings earlier than expected. I will make contributions to both the control and graphic assignment ad hoc groups. My graphic character assignment recommendations address three distinct areas which are related. The areas are the character set, the specific assignment in the code chart and finally, the keyboard layout. Figures 1, 2 and 3 attached reflect my current thinking. I still have reservation about some of the character assignments and how they will be accessed from the keyboard.

> Character Set: The character set that I prefer is attached as Figure 1. As in all previous proposals the character set includes those necessary for the generation of the APL language. In addition, the seven fraction characters are included. I have not included the four card symbols spade, heart, club and diamond. I did not feel that these symbols would obtain general acceptance. You will note that the four cloud symbols have been added and one symbol location remains open. Finally, I am not completely satisfied on the position of the "corners".
>
> Character assignments on right half of 8-bit code chart plus 7/15: My specific assignments are found in Figure 2. The most significant item here is the lack of collating sequence for the lower case Greek alphas. The major reason for this is the layout that occurs when the character set is applied to the keyboard. The operator interface is a very important link in data transfer. I firmly believe that from a system standpoint, the few seconds (microseconds?) wasted in the computer collating is more than offset by the hours (seconds?) saved by the operator in thru put efficiency. The phonetic association of the Greek alphabet with the phonetic Latin alphabet provides this efficiency. One concern about the fraction assignments will be discussed below.

NMAH 311/9

Keyboard Assignments: The keyboard assegnments, on a bit paired basis, from the above character assignments are shown in Figure 3. In the assignment of characters to both the keyboard and code chart the operator should be kept in mind. Keyboard layouts based on Human Factors should be given prime consideration. From your earlier document, I have moved the Greek alphabetic ρ (Rho) from the P key to the R key. My concern about the fraction characters is in their case relation on the keyboard. Should they be placed in the shift case as shown or the upper shift case on the keytops? If this proposal were to become the Shift Out Character set the numerals would be displaced. I have not come to any conclusions as yet, but I want you to be aware of the implications. The problem of generating six characters from one keytop presents logistics problems to the operator. Some operators are confused about the operation of the CONTROL key (3 characters on one keytop). How are they going to accept cord organ playing of the SHIFT, CONTROL, EXTEND, EXTEND SHIFT and EXTEND CONTROL? For lack of a better suggestion, I will accept yours of the three keys next to each other. I do feel that it would benefit the operator if the CONTROL key were added to the right side. I do not agree that an EXTEND LOCK key is required as in your proposal.

The above discussion is at least a minimum contribution, but because of time constraints it was impossible to do more. Perhaps when I return we can get together at our Skokie plant and discuss the graphic characters and assignments.

Yours very truly,

Stuart Garland

STUART M. GARLAND
Industry Standards

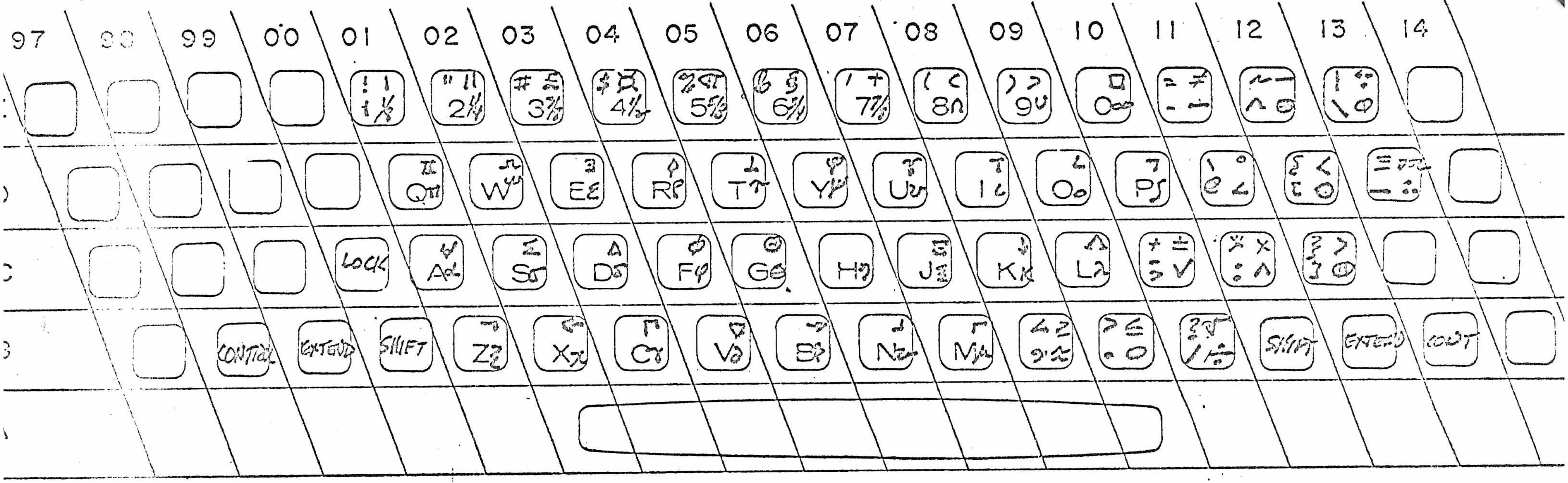
SMG:jb

Att.
Copies to:
Mr. J. B. Booth - 9030
Mr. J. A. Bellino - 9140
Mr. E. H. Clamons ← THIS COPY FOR
Mr. J. J. Tropsa
X4-A9 Keyboard Arrangements (BEMA)
Mr. C. D. Card

J. M. CHALLAND
4/28/71

8 BIT CODE
DRAFT
FIG 2

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0			SP	0	@	P	`	p			□	∞	∠	¬	°	∫
1			!	1	A	Q	a	q			|	1/8	∀	Π	α	π
2			"	2	B	R	b	r			‖	1/4	→	↑	β	ρ
3			#	3	C	S	c	s			£	3/8	Γ	Σ	γ	σ
4			$	4	D	T	d	t			¤	1/2	Δ	⊥	δ	τ
5			%	5	E	U	e	u			π	5/8	∃	ϒ	ε	υ
6			&	6	F	V	f	v			§	3/4	ϕ	∇	φ	∂
7		CONTROLS	'	7	G	W	g	w	CONTROLS	CONTROLS	⊤	7/8	Ⓜ	Ω	θ	ω
8	CONTROLS		(	8	H	X	h	x			⊂	∩		←	η	χ
9			)	9	I	Y	i	y			⊃	∪	⊤	Ψ	ι	ψ
10			*	:	J	Z	j	z			×	∧	⌶	⌉	ξ	ζ
11			+	;	K	[	k	{			±	∨	↓	○	κ	<
12			,	<	L	\	l	¦			≈	≥	Λ	⌽	λ	∵
13			-	=	M	]	m	}			—	≠	⌈	⦷	μ	>
14			.	>	N	^	n	~			○	≤	⌋	⊕	ν	‾
15			/	?	O	_	o	≅			÷	√	⌊	∴	ο	DEL



KBD FOR 8BIT CODE
DRAFT

FIG 3

S M GARLAND
4/28/71

X3L2/1112
May 7, 1971
Project 4
Reply to: C D Card
Site 2-D-3
UNIVAC,
Division of Sperry Rand
P. O. Box 500
Blue Bell. PA 19422

TO: Members, Alternates, Consultants, and Observers of Committee X3L2.

FROM: E. H. Chamens, Chairman

SUBJECT: Letter ballot on candidates for registration

An ad hoc sub-group consisting of John Fletcher, Chairman, E. H. Chamons, and S. Garland was charged by the chairman of the ad hoc group on project 4 to generate the attached proposal.

Please execute the ballot below, and please keep in mind that the technical merit of the proposal, not the registration procedure nor the code extension procedure, is at stake here.

The technical worth of this document is best judged by its merit as an 8-bit code as shown in Appendix A.

Please register my vote

☐ in favor

☐ against (requires written comment)

the proposed candidates for Registration X3L2/1112.

____

Member X3L2

NMAH 311/3

X3L2/1112

Two Candidates for Registration as Character Sets

1. SCOPE.

These character sets constitute an expansion of ASCII by means of code extension to include 190 graphics (in addition to space).

2. PURPOSE

These character sets are introduced to accommodate a greater variety of symbols for use in science and industry and to facilitate their automatic handling in text.

3. EXPLANATION

The Space (2/0) is not included in the sets described here because it is unaffected by shift precedence. This implies that only 94 graphics are replaceable by means of code extension techniques.

A 95th graphic is introduced into each set by the two character escape sequence 1/11, 6/0. This additional character is needed so that two sets can completely fill the available graphic code positions in an 8-bit code table. If a 95-character set is invoked for the left half (columns 02-07) of an 8-bit table, then the 95th character is associated with 07/15(which is unused, since Delete is at 15/15); if it is invoked for the right half (columns 10-15) of an 8-bit table, then the 95th character is associated with 10/0 (which is unused, since space is shift-insensitive).

The intent of this registration is that the two sets be used together in an 8-bit environment.

4. Set #1: The Basic Code.

COLUMN → ROW ↓	2	3	4	5	6	7
0		0	@	P	`	p
1	!	1	A	Q	a	q
2	"	2	B	R	b	r
3	#	3	C	S	c	s
4	$	4	D	T	d	t
5	%	5	E	U	e	u
6	&	6	F	V	f	v
7	'	7	G	W	g	w
8	(	8	H	X	h	x
9	)	9	I	Y	i	y
10	*	:	J	Z	j	z
11	+	;	K	[	k	{
12	,	<	L	\	l	¦
13	−	=	M	]	m	}
14	.	>	N	^	n	~
15	/	?	O	_	o	

1/11, 6/0
≡

Legend: Except for the addition of a 95th character, this set is identical with ASCII as defined in X3.4-1969 and is related to ISO R646. Only the 95th character is described here.

Column/Row	Symbol	Name
1/11, 6/0	≡	equivalence

5. Set #2· The Extended Code.

Row ↓ / Column →	2	3	4	5	6	7
0		∞	§	L	°	ρ
1	|	$\frac{1}{8}$	∀	⌶	α	π
2	‖	$\frac{1}{4}$	⊅	◊	β	∫
3	£	$\frac{3}{8}$	Γ	Σ	γ	σ
4	¤	$\frac{1}{2}$	Δ	⊤	δ	τ
5	¶	$\frac{5}{8}$	∃	ϒ	ε	υ
6	¢	$\frac{3}{4}$	Φ	∇	φ	∂
7	†	$\frac{7}{8}$	⊖	Ω	θ	ω
8	⊂	∩	Ⱥ	⊣	η	χ
9	⊃	∪	∀̸	Ψ	ι	ψ
10	×	∧	Ξ	╪	ξ	ζ
11	±	∨	⊢	←	κ	⟨
12	≈	≥	Λ	↑	λ	⊥
13	—	≠	┘	→	μ	⟩
14	•	≤	┐	⌉	ν	_
15	÷	√	┌	↓	ο	

1/11, 6/0

$\boxed{□}$

Legend:

Column/Row	Symbol	Name
1/11, 6/0	□	quad
2/1	|	stroke
2/2	‖	parallel
2/3	£	Pound Stirling
2/4	¤	scarab (sun, APL circula
2/5	¶	paragraph
2/6	¢	left non-implication (cent)
2/7	†	dagger
2/8	⊂	left implication (subset)
2/9	⊃	right implication (superset
2/10	×	times
2/11	±	plus or minus
2/12	≈	approximates
2/13	—	wide dash
2/14	•	dot (bullet)
2/15	÷	divide

3/0	∞	infinity	4/0	∮	section (loop integral)
3/1	⅛	one-eighth	4/1	∀	universal quantifier (APL locked function)
3/2	¼	one-quarter	4/2	⊅	right non-implication
3/3	⅜	three-eighths	4/3	Γ	upper case gamma
3/4	½	one-half	4/4	Δ	upper case delta
3/5	⅝	five eighths	4/5	∃	existential quantifier
3/6	¾	three-quarters	4/6	Φ	upper case phi (APL rotate)
3/7	⅞	seven-eighths	4/7	Θ	upper case theta (APL log)
3/8	∩	cap (intersection)	4/8	⊼	nand
3/9	∪	cup (union)	4/9	⊽	nor
3/10	∧	and	4/10	Ξ	upper case xi (APL quote qua
3/11	∨	or	4/11	⊢	left edge (assert)
3/12	≥	greater than or equal	4/12	Λ	upper case lambda
3/13	‡	not equal (double dagger)	4/13	⌟	lower right corner
3/14	≤	less than or equal	4/14	⌝	upper right corner
3/15	√	radical (check)	4/15	⌈	upper left corner (maximum)

5/0	⌊	lower left corner (minimum)	6/0	°	degree (circular accent)
5/1	Π	upper case pi (product)	6/1	α	alpha (varies)
5/2	◊	diamond (lozenge)	6/2	β	beta
5/3	Σ	upper case sigma (sum)	6/3	γ	gamma
5/4	⊤	upper edge (APL encode)	6/4	δ	delta
5/5	ϒ	upper case epsilon (APL grade up)	6/5	ε	epsilon (member)
5/6	∇	del (APL function)	6/6	φ	phi (APL transpose)
5/7	Ω	upper case omega (APL comment)	6/7	θ	theta
5/8	⊣	right edge	6/8	η	eta
5/9	Ψ	upper case psi (APL grade down)	6/9	ι	iota (APL index)
5/10	ǂ	exclusive or	6/10	ξ	xi
5/11	←	left arrow (assign)	6/11	κ	Kappa
5/12	↑	up arrow (APL take)	6/12	λ	lambda
5/13	→	right arrow (APL branch)	6/13	μ	mu
5/14	¬	not	6/14	ν	nu
5/15	↓	down arrow (APL drop)	6/15	ο	omicron (APL outer)

7/0	ρ	rho (APL dimension)
7/1	π	pi
7/2	∫	integral
7/3	σ	sigma
7/4	τ	tau
7/5	υ	upsilon
7/6	∂	partial derivative
7/7	ω	omega
7/8	χ	chi
7/9	ψ	psi
7/10	ζ	zeta
7/11	⟨	opening angle.
7/12	⊥	lower edge (perpendicular, APL decode)
7/13	⟩	closing angle
7/14	¯	overbar (APL high minus)

APPENDIX A

Use in an 8-bit Environment.

A1. The Complete Code.

In an 8-bit environment, the graphic sets of this registration are intended to be used as an 8-bit code as shown here, with the basic code on the left and the extended code on the right.

COLUMN → ROW ↓	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	NUL	DLE	SP	0	@	P	`	p			□	∞	§	└	c
1	SOH	DC1	!	1	A	Q	a	q			|	1/8	∀	⌶	α
2	STX	DC2	"	2	B	R	b	r			‖	1/4	⊇	◊	β
3	ETX	DC3	#	3	C	S	c	s	Additional		≐	3/8	Γ	Σ	γ
4	EOT	DC4	$	4	D	T	d	t			⌺	1/2	Δ	⊤	δ
5	ENQ	NAK	%	5	E	U	e	u	Controls		¶	5/8	∃	ϒ	ε
6	ACK	SYN	&	6	F	V	f	v			¢	3/4	Φ	∇	φ
7	BEL	ETB	'	7	G	W	g	w	(To be		†	7/8	Θ	Ω	θ
8	BS	CAN	(	8	H	X	h	x			⊂	∩	⍲	⊣	η
9	HT	EM	)	9	I	Y	i	y	selected)		⊃	∪	⍱	Ψ	ι
10	LF	SUB	*	:	J	Z	j	z			×	∧	Ξ	≢	ξ
11	VT	ESC	+	;	K	[	k	{			±	∨	⊦	←	κ
12	FF	FS	,	<	L	\	l	¦			≈	≥	Λ	↑	λ
13	CR	GS	–	=	M	]	m	}			—	≑	┘	→	μ
14	SO	RS	.	>	N	^	n	~			°	≤	┐	¬	ν
15	SI	US	/	?	O	_	o	≡			÷	√	┌	↓	ο

A2. Keyboard Considerations

A keyboard based on the bit paired version of the proposed standard X4.00-1971 is shown; this keyboard conforms to ECMA-14, type C, version II, and is a candidate for adoption by ISO. While the keyboard shown is designed for direct generation of 8-bit codes, it can also be modified for use in a 7-bit environment: The Extend Key could generate 0/14 when depressed and 0/15 when released; the Quad and Equivalence Keys could generate the necessary two-character escape sequence.

APPENDIX A

Use in an 8-bit Environment.

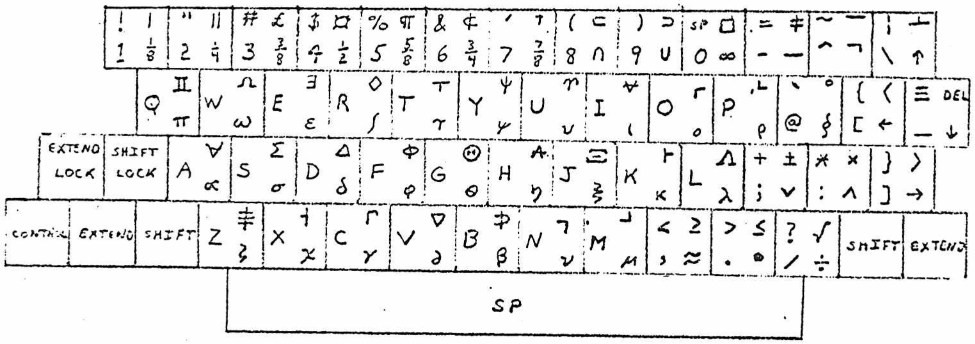
A1. The Complete Code.

In an 8-bit environment, the graphic sets of this registration are intended to be used as an 8-bit code as shown here, with the basic code on the left and the extended code on the right.

COLUMN → ROW ↓	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	NUL	DLE	SP	0	@	P	`	p			□	∞	§	L	°	ρ
1	SOH	DC1	!	1	A	Q	a	q			|	$\frac{1}{8}$	∀	II	α	π
2	STX	DC2	"	2	B	R	b	r			‖	$\frac{1}{4}$	⊅	◇	β	∫
3	ETX	DC3	#	3	C	S	c	s	Additional		£	$\frac{3}{8}$	Γ	Σ	γ	σ
4	EOT	DC4	$	4	D	T	d	t			¤	$\frac{1}{2}$	△	⊤	δ	τ
5	ENQ	NAK	%	5	E	U	e	u	Controls		¶	$\frac{5}{8}$	∃	ϒ	ε	ν
6	ACK	SYN	&	6	F	V	f	v			¢	$\frac{3}{4}$	Φ	∇	φ	∂
7	BEL	ETB	´	7	G	W	g	w	(To be		†	$\frac{7}{8}$	Θ	Ω	θ	ω
8	BS	CAN	(	8	H	X	h	x			⊂	∩	A	⊣	η	χ
9	HT	EM	)	9	I	Y	i	y	selected)		⊃	∪	∀	Ψ	ι	ψ
10	LF	SUB	*	:	J	Z	j	z			×	∧	⌶	⋕	ξ	ζ
11	VT	ESC	+	;	K	[	k	{			±	∨	⊢	←	κ	⟨
12	FF	FS	,	<	L	\	l	¦			≈	≥	Λ	↑	λ	⊥
13	CR	GS	−	=	M	]	m	}			—	≠	┘	→	μ	⟩
14	SO	RS	.	>	N	^	n	~			•	≤	┐	¬	ν	—
15	SI	US	/	?	O	_	o	≣			÷	√	┌	↓	ο	DE_

A2. Keyboard Considerations

A keyboard based on the bit paired version of the proposed standard X4. 00-1971 is shown; this keyboard conforms to ECMA-14, type C, version II, and is a candidate for adoption by ISO. While the keyboard shown is designed for direct generation of 8-bit codes, it can also be modified for use in a 7-bit environment: The Extend Key could generate 0/14 when depressed and 0/15 when released; the Quad and Equivalence Keys could generate the necessary two-character escape sequence.



Depression of the Extend Key asserts the eighth bit. When used with the shift key, it behaves as follows: (1) If neither Shift nor Extend is depressed, the character at the lower left of the Keytop (or a lower case latin letter) is selected. (2) If only Shift is depressed, the character at the upper left of the Keytop (or an upper case Latin letter) is selected. (3) If only Extend is depressed, the character at the lower right of the Keytop is selected. (4) If both shift and Extend are depressed, the character at the upper right of the Keytop is selected. Color-coding could be an aid to the operator. The Extend Key might also have use in connection with the Control Key.

APPENDIX B

Design Considerations for the Character Sets.

B1. Introduction.

These graphic sets are intended to be used together, the combined sets constituting a character set suitable for a wide range of applications relating to business and/or scientific pursuit.

B2. Considerations Affecting the Code.

B2. 1. Characters were selected to make the combined sets suitable for an environment in which the same communication and/or data processing equipment is used by persons having interests in some or all of the following areas:

(1) business, commerce, and finance,
(2) science, mathematics, and technology,
(3) programming languages (specifically, in addition to those accommodated by the standard set, Algol and APL),
(4) tabular data presentation.

It is realized that the code cannot satisfy all specialized applications in these areas.

B2. 2. The environment noted in B2. 1 clearly suggests employment of an 8-bit code. Accordingly, the registered sets contain 95 characters each.

B2. 3. Code assignments were made to preserve a close relationship between pairs of characters having the same position in the extended set and the basic set (which are here called matched characters). Close relationship was judged by

(1) mnemonic considerations,
(2) placement on the standard (bit-paired) Keyboard,
(3) international considerations.

The typewriter paired keyboard was not considered, since that keyboard is intended for trained touch typists who do not need mnemonic clues as badly as would those who use the bit-paired keyboard for general purpose and program entry.

B2.4. That the basic set should conform to the Standard Code of X3.4-1969 was accepted a priori.

B3. Choice of Graphics

B3.1. Those upper case Greek letters are included which have a printed form readily distinguishable from Latin letters. All eleven of these characters appear in scientific and mathematical writing. They are placed in code table columns 4 and 5 (as are the upper case Latin letters) in positions associated with the corresponding lower case Greek letters.

B3.2. The full 24-character lower case Greek alphabet is included. Nearly all of these appear in scientific and mathematical writing. The only one with a printed form not necessarily distinguishable from Latin letters is omicron; this is retained for completeness and because of its applicability to APL. If a lower case Greek letter shares its upper case form with a Latin letter, the Greek letter is given the same code table placement as the lower case Latin letter. In particular rho, rather than pi, must be matched with p; to reduce confusion on this point, no Greek letter is matched with r. Eight of the remaining eleven lower case Greek letters are placed because of obvious phonic (delta, phi, lambda, sigma, upsilon) or visual (theta, omega, psi) associations with the matching Latin letters. Pi is matched with q to keep it near p, and xi is matched with j arbitrarily, since there is no natural match (x being already taken). Gamma is matched with c, rather than g, to keep its familiar third position in the collating sequence. Apart from this, the standard Greek collation is ignored since this set set is not intended to be normally used for text in the Greek language; the Greek letters are intended to be used primarily as specialized symbols.

B3.3. The symbols integral and partial derivative, which have a Greek appearance, are placed so to be matched with the remaining two Latin lower case letters (r and v). In the upper case position corresponding to partial derivative is the del (inverted delta), which has a visual resemblance to V, with which it is matched. All three of these characters are widely used in mathematics.

B3.4. The eighth fractions from one-eighth to seven-eighths are placed in code positions occupied by the digits one to seven of the basic set. These have commercial use. A similar placement is that of the mathematical symbol infinity, which is matched with zero.

B3.5. The symbols stroke, Pound Sterling, scarab, section, not, and overbar are all alternate American or international forms of the characters with which they are matched. the alternate uses of these characters are deemed important enough to merit codes of their own in an expanded set. Similarly, the wide dash is introduced, distinct from the hyphen or minus, but matched with it.

B3.6. Arrows in all four directions are included and placed so as to appear on a cluster of adjacent keys at the right side of the standard keyboard. Further, left and right arrows are matched with brackets of similar sense. In the same way, angle brackets are introduced and matched with the corresponding braces. These six symbols are widely used in programming languages and scientific work.

B3.7. Additional valuable mathematical symbols are included and matched with symbols of related meaning or appearance (times, plus or minus, dot, divide, greater than or equal, not equal, less than or equal). In particular, asterisk is used to mean times in Fortran. Also, since not equal is best matched with equal, greater than or equal (not less than) is matched with less than, and less than or equal (not greater than) with greater than. Radical and approximates are placed so as to be near these other mathematical symbols.

B3. 8. Additional valuable typographic symbols are included (parallel, paragraph, dagger) and placed so as to be in a commerically oriented portion of the table, although the symbols have other uses also.

B3. 9. The complete set of ten binary boolean operators is included. Their placement makes some effort to achieve visual similarity with the matching characters of the basic set; the most compelling match is that of the implications with the parentheses. One non-implication is also useable as cent and is placed in a commerical portion of the table. Equivalence is placed as the 95th character of the basic set, since its visual form could make it useful also as a visible delete character for OCR. The two quantifiers are included and matched with similar Latin upper case letters. Cup and cap, which together with the implications form a set of U-shaped symbols facing all four ways are placed near the implications. These symbols are all of use in mathematics or technology.

B3. 10. To facilitate the outlining of tabular data, eight additional plotting symbols are included. It is intended that table outlines pass through the centers of character positions; therefore, stroke, wide dash, and plus can be used for vertical edges, horizontal edges, and cross-overs respectively. The eight additional symbols enable the formation of corners (four L-shaped symbols), and edge intersections (four T-shaped symbols).

B3. 11. The table is completed with degree (matched with grave accent, since both are small and elevated), quad, and diamond. Quad is the 95th character of the extended set and thus in some sense is correlated with Space; it can be interpreted as a visible space.

B4. Comment on Programming Languages.

B4. 1. The legend indicates how the set will accommodate the more exotic APL graphics. In addition, the (broken) vertical of the basic set is assumed as an acceptable I-beam. Some variations from the usual APL printed form are required; such variations are minor, except for characters obtained by over-strike in the original APL implementation. It seems that further accommodation of APL in a set intended for general use is not possible, since many APL characters in their original form have no other use.

B4. 2. If a program written in the 190-character set of this registration must be used in an environment in which only the standard set is available, suitable translation algorithms can be defined without difficulty.